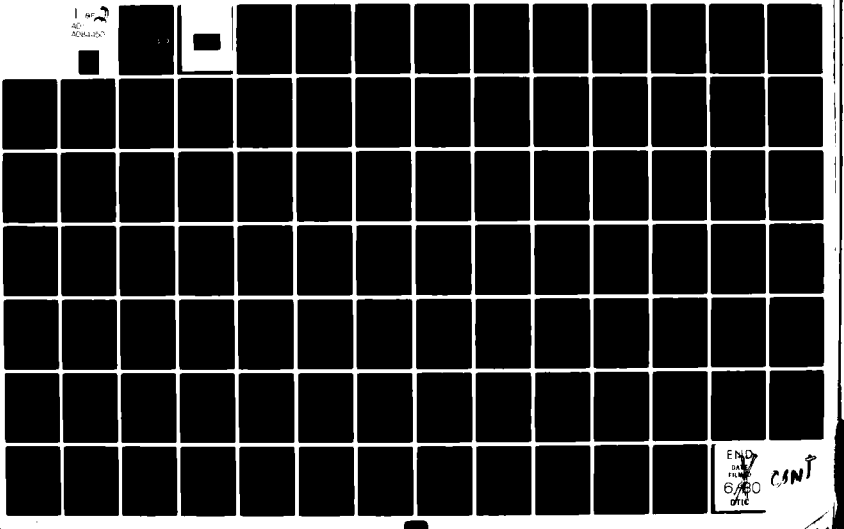
AD-A084 450 BROWN UNIV PROVIDENCE R I DIV OF APPLIED MATHEMATICS F/6 20/4
FINITE ELEMENT METHODS FOR HEAT TRANSFER PROBLEMS.(U)
APR 80 F BISSHOPP, R B CASWELL, M E MICHAUD DAAG29-76-G-0178
UNCLASSIFIED ARO -13548.2-MX NL



END
DATE FILMED
6-80
DTIC

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER (19) 13548.2-MX (18) ARO	2. GOVT ACCESSION NO. AD-A084450
3. RECIPIENT'S CATALOG NUMBER	
4. TITLE (and Subtitle) (6) Finite Element Methods for Heat Transfer Problems,	5. TYPE OF REPORT & PERIOD COVERED (9) Final Report 15 Jun 76 — 15 Sep 79
	6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) (10) F. Bisshopp, R. B. Caswell, M. E. Michaud	8. CONTRACT OR GRANT NUMBER(s) DAAG29-76-G-0178 (15)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Brown University, Providence, RI, 02912	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS (12) 94
11. CONTROLLING OFFICE NAME AND ADDRESS U. S. Army Research Office, Post Office Box 12211, Research Triangle Park, NC 27709	12. REPORT DATE (11) 2 Apr 80
	13. NUMBER OF PAGES 88
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)	15. SECURITY CLASS. (of this report) Unclassified
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

DTIC ELECTE MAY 21 1980 S D A

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

NA

18. SUPPLEMENTARY NOTES

The view, opinions, and/or findings contained in this report are those of the author(s) and should not be construed as an official Department of the Army position, policy, or decision, unless so designated by other documentation.

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

two-phase flow
porous media
waterflood
mesh refinement

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

Research on two-phase flow in porous media carried out in the Division of Applied Mathematics at Brown University

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

ADA084450

FINITE ELEMENT METHODS

FOR HEAT TRANSFER PROBLEMS

Final Report and
Appendices A, B, C, D, F

DAAG29-76-G-0178

FINITE ELEMENT METHODS
FOR HEAT TRANSFER PROBLEMS

FINAL REPORT

2 April 1980

U.S. Army Research Office

Grant No. DAAG29-76-G-0178

Brown University
Providence, Rhode Island 02912

Foreword

The first seven sections of this report contain brief, but technical descriptions of research on numerical fluid mechanics that has been carried out at Brown University under contract with the Army Research Office. More detailed descriptions of parts of that research are provided in six appendices.

The sections, except Section 8, are arranged in chronological order according to when we began work on the various problems. At the start, three and a half years ago, the finite fluid element method (Section 1) was the only one under consideration. Things went badly with that method, and progress with its implementation went far more slowly than we ever anticipated. As a result, by the end of the first year, a second method, which originally was developed for a check on results of the first, had become by far the more promising of the two. Most of this report (Sections 3-6) describes progress we have made with the application of biased differences (Section 2) to a variety of fairly difficult problems of numerical fluid mechanics. Finally, in the last six months of the period covered by this report, the major difficulties with finite fluid elements were overcome, so it became possible to begin a comparison of the two methods (Section 6). Preliminary indications are that both methods are reliable, and both are considerably more efficient than a third method with which they have been compared.

Section 7 is a report of progress with a boundary integral method that is not closely related to the others except by being numerical, and Section 8 is a description of the kinds of graphical software we had to develop for interpretation of our numerical computations.

The research reported here has been carried out by F. E. Bisshopp, R. B. Caswell (principal investigators), E. W. Fleri, M. E. Michaud, and T. G. McKee (research assistants in Applied Mathematics).

Table of contents

List of Appendices

1. Finite fluid elements.

As initially formulated, the finite fluid element method dealt with localized distributions of mass, momentum and energy. The mass density and momentum density of a fluid were represented as

$$\rho(\underline{x},t) \sim \sum_1^N m_i(t) f_i(\underline{x} - \underline{X}_i(t))$$

$$\rho u(\underline{x},t) \sim \sum_1^N m_i \dot{\underline{X}}_i f_i(\underline{x} - \underline{X}_i)$$

with distribution functions in n = 1, 2 or 3 dimensions.

$$f_i = \frac{1}{(\pi\sigma_i^2)^{n/2}} e^{|\underline{x}-\underline{X}|^2/\sigma_i^2}$$

The equations of fluid mechanics were then used, along with least square fitting of the approximations, to obtain ordinary differential equations for the fluid element parameters $m_i(t)$, $\underline{X}_i(t)$ and $\sigma_i(t)$. A detailed description of the method is included in appendix A: here we will report on two major difficulties and what has been developed to overcome them.

In the course of our attempts to implement a finite fluid element method we found first that the original formulation was simply too complicated to be of practical value -- it was never successfully employed for a two-dimensional flow. Instead, two-dimensional flow has been treated by an algorithm that is based upon the original one, but is greatly simplified as follows:

Simulation of the equation of continuity has been dropped in favor of fixed particle masses, thus bringing the method somewhat nearer to a particle-in-cell method. The basic approximation is

$$\rho(\underline{x},t) \sim \sum_1^N m_i f_i(\underline{x} - \underline{X}_i(t))$$

where the integral of f_i over all of 1, 2 or 3-dimensional space is

$$\langle f_i \rangle = 1.$$

The mass density at the center of the i^{th} element is assigned the (approximate) value

$$\langle \rho f_i \rangle = \sum_1^N m_j \langle f_i f_j \rangle = \rho_i$$

where

$$\langle f_i f_j \rangle = \frac{1}{(\pi(\sigma_i^2 + \sigma_j^2))^{n/2}} e^{-|\underline{X}_i - \underline{X}_j|^2/(\sigma_i^2+\sigma_j^2)}$$

and the parameters σ_i that characterize element diameters are adjusted to give an overlap of the i^{th} element with 2, 6 or 12 neighbors in 1, 2 or 3 dimensions. Some indication of ways to assign the σ's is given in appendix A, and further investigation of the effect of that choice on accuracy of the method is still in progress.

Motion of the centers of the elements is governed by the approximation of the fluid momentum equation,

$$\rho_i \ddot{\underline{X}} = \rho_i \underline{g}(\underline{X}_i) - (\underline{\nabla} p)_i + (\underline{\nabla} \cdot \tau)_i$$

for viscous flow, or

$$\rho_i \ddot{\underline{X}}_i = \rho_i \underline{g}(\underline{X}_i) - (\underline{\nabla} p)_i - \mu(\underline{X}_i)\dot{\underline{X}}_i$$

for flow in a porous medium. The body force $\underline{g}(\underline{x})$ has presented no difficulty, and, for a barotropic fluid, it has been found that an adequate approximation of the pressure gradient is

$$(\underline{\nabla} p)_i = p'(\rho_i)(\underline{\nabla} \rho)_i$$

where

$$(\underline{\nabla}\rho)_i = \sum_1^N m_j \underline{\nabla} \langle f_i f_j \rangle$$

$$= 2 \sum_1^N \frac{m_j}{\sigma_i^2+\sigma_j^2} (\underline{X}_j - \underline{X}_i) \langle f_i f_j \rangle .$$

A second difficulty with the initial formulation was found in the attempt to approximate the effect of viscous stress The quadratic fitting of velocities outlined in appendix A gave a rather poor estimate of the second gradients that appear in a direct evaluation of $(\nabla\cdot\tau)_i$. It has been found now that a much better representation of viscous effects can be obtained in terms of two estimates of first gradients. In the simplified algorithm for viscous flow there is a viscous stress field

$$\tau(\underline{x},t) \sim \sum_1^N t_i f_i(\underline{x} - \underline{X}_i(t)) ,$$

and its divergence is approximated, similarly to $\underline{\nabla}\rho$, as

$$(\underline{\nabla}\cdot\tau)_i = 2 \sum_1^N \frac{(\underline{X}_j-\underline{X}_i)\cdot t_j}{\sigma_i^2 + \sigma_j^2} \langle f_i f_j \rangle \quad .$$

The coefficients t_i are defined implicitly by approximate values of the stress field as

$$\tau_i = \sum_1^N t_j \langle f_i f_j \rangle$$

where

$$\tau_i = \lambda(\rho_i)(\underline{\nabla}\cdot\underline{u})_i I + \mu(\rho_i)((\underline{\nabla}\,\underline{u})_i + (\underline{\nabla}\,\underline{u})_i^T)$$

and $(\underline{\nabla}\,\underline{u})_i$ is the outer product

$$(\underline{\nabla}\,\underline{u})_i = 2 \sum_1^N \frac{(\underline{X}_j-\underline{X}_i)}{\sigma_i^2 + \sigma_j^2} \dot{\underline{X}}_j \langle f_i f_j \rangle \quad .$$

Investigation of performance of the 1 and 2-dimensional versions of the simplified algorithm is still in progress. Results on sound waves and shock waves will appear in the Ph.D. dissertaion of E. W. Fleri, and he is scheduled to present a poster session on the subject at the SIAM Meeting, Alexandria, Va., 5-7 June 1980.

2. Biased differences.

The method of biased differences was originally formulated and developed as a standby -- it was to be used for independent checks of results of the finite fluid element method. As it turned out, however, very few problems with it were encountered, and its present state of development is considerably advanced beyond that of the finite fluid element method. It has been applied, with promising results, to relatively difficult problems of 2 and 3-dimensional unsteady flow. In this section, we give a detailed description of the application of the method to the simple problem of Burgers' equation in one dimension.

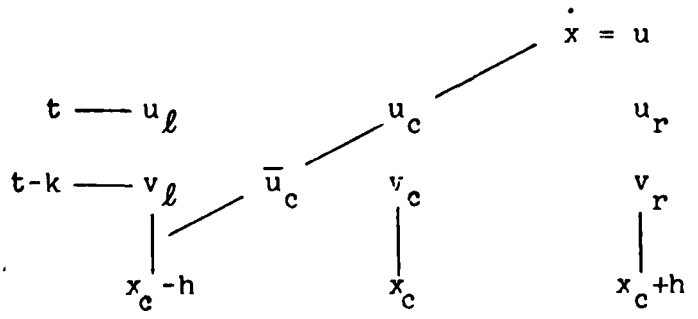
Given

$$u_t + uu_x = \nu u_{xx},$$

the left-hand-side is a material derivative, i.e. the total time derivative along a particle path defined by $\dot{X} = u(X(t),t)$, and the mixed Eulerian-Lagrangian formulation of the Burgers' equation is

$$\dot{X} = u$$
$$\dot{u} = \nu u_{xx} .$$

By contrast with finite fluid elements that move along particle paths, biased differences are defined on a grid that is fixed in space and has the local skeleton,



There are several possibilities for defining the bias line (approximate particle path in this case) and performing the

time integration to various orders of accuracy. By experiment it has been found that the following relatively simple, low order scheme gives quite reliable results.

1) $\bar{u}_c$ is taken to be the linear interpolation of v_ℓ(or v_r) and v_c at a point $\bar{x}_c$

$$\bar{u}_c = v_c \frac{(\bar{x}_c - x_\ell)}{h} + v_\ell \frac{(x_c - \bar{x}_c)}{h} \qquad (v_c \geq 0)$$

$$\bar{u}_c = v_c \frac{(x_\ell - \bar{x}_c)}{h} + v_r \frac{(\bar{x}_c - x_c)}{h} \qquad (v_c \leq 0)$$

2) The bias line is defined by the forward difference $x_c = \bar{x}_c + k\bar{u}_c$, and elimination of $x_c - \bar{x}_c$ gives

$$\bar{u}_c = \frac{v_c}{1 - \frac{k}{h}(v_c - v_\ell)} \qquad (v_c \geq 0)$$

$$\bar{u}_c = \frac{v_c}{1 + \frac{k}{h}(v_r - v_c)} \qquad (v_c \leq 0)$$

3) The equation for $\dot{u}$ is approximated by a backward difference in time and a central difference in space to give

$$u_c - \frac{\nu k}{h^2}(u_\ell + u_r - 2u_c) = \bar{u}_c \quad .$$

The low order integration scheme has a local truncation error of $O(k^2)$ in its time integrations, but it has <u>not</u> been found necessary to improve that by introduction of central differences. An indication of how the combined forward and backward differences behave like central differences is given in appendix B.

3. Two-dimensional unsteady flow

Introduction of body forces and application of biased differences to 2 and 3-dimensional versions of Burgers' equation is straightforward and will not be discussed here. We turn now to the more difficult problems of incompressible fluid mechanics. In mixed Eulerian-Lagrangian form the equations of motion are

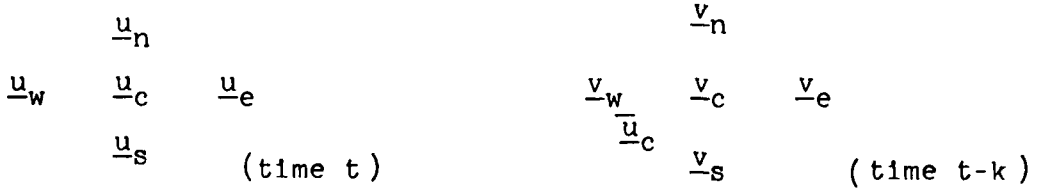
$$\dot{\underline{X}} = \underline{u}$$

$$\dot{\underline{u}} = \underline{f} - \underline{\nabla}\pi + \nu\Delta\underline{u}$$

$$\underline{\nabla}\cdot\underline{u} = 0$$

where $\underline{f}$ is a specified body force, π is p/ρ , and ν is the kinematic viscosity.

As in the one-dimensional example, the grid is fixed in space, $\underline{u}_c$ refers to a central node at time t, $\underline{v}_c$ refers to the same node at time t-k, $\underline{x}_c$ is the position of a node, and $\bar{\underline{x}}_c$ is the position at time t-k of the bias line that passes through $\underline{x}_c$ at time t. The local skeleton (in plan views) is



The forward difference approximation,

$$x_c = \bar{x}_c + k\bar{\underline{u}}_c$$

and linear interpolation for $\bar{\underline{u}}_c$ now gives simultaneous linear equations for the two (or three) components of $\bar{\underline{u}}_c$. The equations depend upon which quadrant (or octant) contains $\bar{\underline{x}}_c$ and that is decided by the signs of the components of $\underline{v}_c$. If both components of $\underline{v}_c$ are positive, for example, $\bar{\underline{x}}_c$ is in the third quadrant and, with subscripts 1 and 2 to denote components of $\bar{\underline{u}}$ and $\underline{v}$,

$$(1 + \frac{k}{h}(v_{1c} - v_{1e}))\bar{u}_{1c} + \frac{k}{h}(v_{1c} - v_{1s})\bar{u}_{2c} = v_{1c}$$

$$\frac{k}{h}(v_{2c} - v_{2e})\bar{u}_{1c} + (1 + \frac{k}{h}(v_{2c} - v_{2s}))\bar{u}_{2c} = v_{2c} \ .$$

This gives initial values, $\bar{u}$, for the integration of the momentum equation.

To find the velocity field, the momentum equation, is first approximated by a backward time difference with no space differences included yet:

$$\underline{u}_c = \bar{\underline{u}}_c + k(\underline{f}(\underline{x}_c) - (\underline{\nabla}\pi)_c + \nu(\Delta\underline{u})_c) \ .$$

Then the central difference approximation of $\underline{\nabla}\cdot\underline{u} = 0 = \Delta\underline{\nabla}\cdot\underline{u}$ gives the pressure equation,

$$\pi_u + \pi_s + \pi_e + \pi_w - 4\bar{\pi}_c = h(\frac{f_1(\underline{x}_e) - f_1(\underline{x}_w)}{2} + \frac{f_2(\underline{x}_n) - f_2(\underline{x}_s)}{2} + \frac{h}{k}(\frac{\bar{u}_{1e} - \bar{u}_{1w}}{2} + \frac{\bar{u}_{1n} - \bar{u}_{1s}}{2})$$

With values of π determined by approximate solution of the Poisson equation, central space differences in the time-integrated momentum equation finally give

$$u_{1c} - \frac{\nu k}{h^2}(u_{1n} + u_{1s} + u_{1e} + u_{1w} - 4u_{1c}) = \bar{u}_{1c} + kf_1(\underline{x}_c) + \frac{k}{h}\frac{\pi_e - \pi_w}{2}$$

$$u_{2c} - \frac{\nu k}{h^2}(u_{2n} + u_{2s} + u_{2e} + u_{2w} - 4u_{2c}) = \bar{u}_{2c} + kf_2(\underline{x}_c) - \frac{k}{h}\frac{\pi_n - \pi_s}{2} \ .$$

In practice, it has been found that the implicit equations for π and $\underline{u}$ at the nodes can be treated quite efficiently by iterations, with starting values obtained from data at time t-k.

There remains, as always, the treatment of boundary conditions. So far, only rectangular geometries with boundaries parallel to a coordinate axis have been considered. At a southern boundary, for example, with $\underline{x}_c$ on the boundary $\underline{\nabla}\cdot\underline{u} = 0$ implies

$$u_2(x_{1c},\ x_{2c} + y) \sim u_2\,(y/h)^2$$

and

$$\nu\Delta u_2 \sim \frac{2\nu}{h^2}\,u_{2n}\,.$$

At $y = \frac{1}{2}h$ the normal component of the momentum equation then gives the pressure boundary condition,

$$\pi_c = \pi_u - \tfrac{1}{2}h(f_2(\underline{x}_n) + f_2(\underline{x}_c)) - \frac{2\nu}{h}\,u_2 + \tfrac{1}{4}\,\frac{h}{k}(u_{2n} - \bar{u}_{2n}).$$

This too is implicit, and iteration has proved effective.

Appendix C is a description of an application of the method to flow in a channel. The results obtained there compared very well with results obtained by a far more elaborate finite element method.

4. Thermal convection

At this time the only three-dimensional unsteady flow that has been simulated is thermal convection. The Boussinesq approximation for nearly incompressible fluids (i.e. liquids) is governed by

$$\dot{\underline{X}} = \underline{u}$$

$$\dot{\underline{u}} = - \underline{\nabla}\pi - \alpha(T - \langle T \rangle)\underline{g} + \nu\Delta\underline{u}$$

$$\dot{T} = \kappa\Delta T + \Phi$$

$$\underline{\nabla}\cdot\underline{u} = 0$$

where

π is $\frac{p}{\rho_o} + V$ --------- nearly constant density

$\underline{g}$ is $- \underline{\nabla} V$ -- gravitational force and potential

α is the coefficient of thermal expansion

$\langle T \rangle$ is the average temperature

ν is the kinematic viscosity

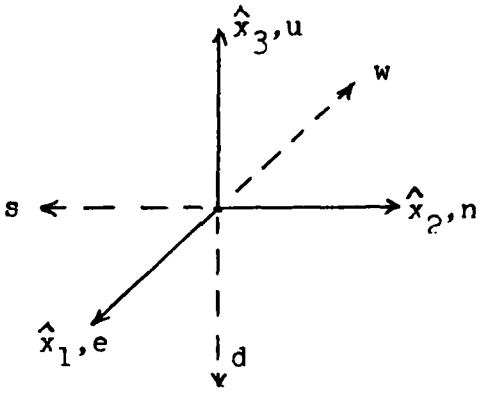
κ is the thermal diffusivity

the viscous dissipation, Φ, is usually negligible.

In essence, the approximation is derived by neglecting compressibility everywhere but in the part of buoyancy that is not derivable from a potential .

The biased difference scheme for these problems is a forward difference of $\dot{\underline{X}} = \underline{u}$ to define a bias line followed by backward differences for the momentum and energy equations, and again

there is a Poisson equation for the pressure. Since initial data $\underline{\bar{u}}_c$ and $\bar{T}_c$ is used, it is preferable to solve the forward difference equations for $\underline{x}_c - \underline{\bar{x}}_c$ instead of $\underline{\bar{u}}_c$. With directions relative to central nodes defined as



the equations for displacements when the components of $\underline{v}_c$ are all positive are

$$(1 + \frac{k}{h}(v_{1c}-v_{iw}))\delta_1 + \frac{k}{h}(v_{1c}-v_{1s})\delta_2 + \frac{k}{h}(v_{1c}-v_{1d})\delta_3 = kv_{1c}$$

$$\frac{k}{h}(v_{2c}-v_{2w})\delta_1 + (1 + \frac{k}{h}(v_{2c}-v_{2s}))\delta_2 + \frac{k}{h}(v_{2c}-v_{2d})\delta_3 = kv_{2c}$$

$$\frac{k}{h}(v_{3c}-v_{3w})\delta_1 + \frac{h}{k}(v_{3c}-v_{3s})\delta_2 + (1 + \frac{k}{h}(v_{3c}-v_{3d}))\delta_3 = kv_{3c}$$

where $\underline{\delta}$ is $\underline{x}_c - \underline{\bar{x}}_c$. Similar sets of equations define $\underline{\delta}$ in the other octants.

Given $\underline{\delta}$ as above, the initial data for the backward differences is

$$\underline{\bar{u}}_c = \underline{v}_c - (\underline{v}_c - \underline{v}_w)\frac{\delta_1}{h} - (\underline{v}_c - \underline{v}_s)\frac{\delta_2}{h} - (\underline{v}_c - \underline{v}_d)\frac{\delta_3}{h}$$

$$\bar{T}_c = T_c - (T_c - T_w)\frac{\delta_1}{h} - (T_c - T_s)\frac{\delta_2}{h} - (T_c - T_d)\frac{\delta_3}{h}$$

and the backward equations are:

$$T_c - \frac{\kappa k}{h^2}(T_e + T_w + T_n + T_s + T_u + T_d - 6T_c) = \bar{T}_c$$

$$\underline{u}_c - \frac{\nu k}{h^2}(\underline{u}_e + \underline{u}_w + \underline{u}_n + \underline{u}_s + \underline{u}_u + \underline{u}_d - 6\underline{u}_c)$$

$$= \bar{\underline{u}}_c + k[(\underline{\nabla}\pi)_c - \alpha \underline{g}(T_c - \langle T \rangle)]$$

$$\frac{1}{h^2}(\pi_e + \pi_w + \pi_n + \pi_s + \pi_u + \pi_d - 6\pi_c) = (\underline{\nabla} \cdot \bar{\underline{u}})_c - \alpha(\underline{g} \cdot \underline{\nabla} T)_c.$$

Some results on thermal instability of a fluid heated from below (the Bénard problem) are given in appendix D.

5. Two phase flow

A considerable effort was devoted to the formulation of a biased difference algorithm for flow of two immiscible liquids in a porous medium. In its full generality the problem there is rather complicated, but there is nothing in its structure that precludes an application of the method to it. For the most part, however, we have concentrated on a simplified version where the sum of the volumetric flow rates of the 'wetting' and 'nonwetting' phases

$$\underline{u} = \underline{u}_w + \underline{u}_n$$

is specified in advance to be an irrotational, incompressible flow field. Then the ratio of the proportions of void volume filled by the phases (saturation)

$$s(\underline{x},t) = V_w/(V_n + V_w)$$

is governed by the saturation equation,

$$\frac{\partial s}{\partial t} + F'(s)\underline{u}\cdot\underline{\nabla}s = \underline{\nabla}\cdot\mathcal{D}(s)\underline{\nabla}s.$$

In this case the forward difference for the bias line,

$$\underline{x}_c = \overline{\underline{x}}_c + kF'(s(\overline{\underline{x}}_c))\overline{\underline{u}}_c \ .$$

is not so easily solved as before because the derivative of the fractional flow rate, $F'(s)$, is <u>not</u> a linear function of s. In fact,

$$F'(s) \simeq \frac{s(1-s)}{(s^2+(1-s)^2)^2}$$

is more typical of the kind of problems encountered in waterflooding of oil producing reservoirs.

The case where F'(s) is given as above and ρ is constant has been investigated in detail by Marion C Michaud in her Ph D dissertation. The dissertation is included here as appendix E: a shorter version for publication is being prepared

6. Compressible flow

A start has been made on the application of biased differences to compressible flow. In the simple barotropic case

$$p = \rho^2$$

$$\dot{\underline{X}} = \underline{u}$$

$$\dot{\rho} = -\rho \underline{\nabla} \cdot \underline{u}$$

$$\dot{\underline{u}} = -2\underline{\nabla}\rho + \frac{1}{\rho}\,\underline{\nabla}\cdot\mu(\underline{\nabla}\,\underline{u} + \underline{\nabla}\,\underline{u}^T - \frac{2}{3}\,\underline{\nabla}\cdot\underline{u}\,\mathbf{I}).$$

Comparisons of results in the one dimensional case have been made between the finite fluid element method, a biased central difference algorithm, and the biased forward-backward difference algorithm.

The central difference algorithm is

$$x_c = \bar{x}_c + \tfrac{1}{2}k(\bar{\rho}_c + \rho_c)$$

$$\rho_c = \frac{\bar{\rho}_c\,(1 - \frac{1}{2}k(\overline{\nabla u})_c}{(1 + \frac{1}{4}\frac{k}{h}(u_r - u_\ell))}$$

$$u_c = [\bar{u}_c - \tfrac{1}{2}k((\overline{\nabla\rho})_c + \frac{(\rho_r - \rho_\ell)}{2h}\,)$$

$$+ \tfrac{1}{2}\nu k((\overline{\Delta u})_c + \frac{1}{h^2}(u_\ell + u_r))]/(1 + \frac{\nu k}{h^2})$$

where bars denote linear interpolation at $\bar{x}_c$. Iteration is carried out on all three equations in the sequence indicated above.

The much simpler forward-backward algorithm is

$$x_c = \bar{x}_c + k\bar{\rho}_c$$

$$\rho_c = \bar{\rho}_c/(1 + \frac{k}{2h}(u_r - u_\ell))$$

$$u_c = [\bar{u}_c - \frac{k}{2h}(\rho_r - \rho_\ell) + \frac{\nu k}{h^2}(u_r + u_\ell)]/(1 + \frac{2\nu k}{h^2})$$

in which only the last two equations are iterated.

The forward-backward finite fluid element method that was used is

$$m_i = 1$$

$$X_i = \bar{X}_i + k\bar{u}_i$$

$$\sigma_i^2 = \tfrac{1}{4} \sum{}'(X_j - X_i)^2 \qquad \text{(2 neighbors)}$$

$$\rho_i = \sum \langle f_i f_j \rangle \qquad \text{(1 and 2 neighbors)}$$

$$(\nabla\rho)_i = 2 \sum{}' \frac{(X_j - X_i)}{\sigma_j^2 + \sigma_i^2} \langle f_i f_j \rangle \qquad \text{(2 neighbors)}$$

$$(\nabla u)_i = 2 \sum{}' \frac{(X_j - X_i)}{\sigma_j^2 + \sigma_i^2} u_j \langle f_i f_j \rangle \qquad \text{(2 neighbors)}$$

$$t_i \langle f_i f_i \rangle = (\nabla u)_i - \sum{}' t_j \langle f_i f_j \rangle$$

$$u_i = \bar{u}_i - 2k(\nabla\rho)_i + \frac{2\mu k}{\rho_i} \sum{}' \frac{(X_j - X_i)}{\sigma_j^2 + \sigma_i^2} t_j \langle f_i f_j \rangle$$

in which the last three equations are iterated.

In simulations of one-dimensional sound and shock waves, all three methods worked well; and we were led to conclude that central time differences give no significant improvements over the simpler methods. Computation times for the forward-backward biased difference and finite fluid element methods were comparable and significantly shorter than corresponding times for central differences.

This work will appear in a Ph.D. dissertation by E. W. Fleri (in preparation), and the two-dimensional version of the finite fluid element method will also appear there.

7. Vortex motion

In some problems of fluid mechanics, there is a boundary integral method that can be used to formulate efficient numerical methods. A particular example is the flow induced by a two-dimensional region of constant vorticity, i.e.

$$\frac{\partial v}{\partial x} - \frac{\partial u}{\partial y} = \omega \qquad \text{in R bounded by C}$$
$$= 0 \qquad \text{outside R}$$

The solution of this problem for which $\underline{u} \to 0$ as $|\underline{x}| \to \infty$ is

$$u(X,Y,t) = \frac{\omega}{2\pi} \int_R \frac{y - Y}{|\underline{x} - \underline{X}|^2} \, dxdy$$

$$v(X,Y,t) = -\frac{\omega}{2\pi} \int_R \frac{x - X}{|\underline{x} - \underline{X}|^2} \, dxdy$$

and this, in turn, can be transformed to the line integral

$$\underline{u}(\underline{X},\tau) = -\frac{\omega}{2\pi} \oint_C \ell n |\underline{x} - \underline{X}| d\underline{x} \, .$$

For two-dimensional, incompressible flow $d\omega/dt = 0$ on particle paths, so the motion of the boundary of R is governed by

$$\dot{\underline{X}} = -\frac{\omega}{2\pi} \oint_C \ell n |\underline{x} - \underline{X}| d(\underline{x}-\underline{X}) \, , \qquad \underline{X} \text{ on C.}$$

The shift of the integration variable from $\underline{x}$ to $\underline{x}-\underline{X}$ removes the logarithmic singularity, for after an integration by parts on the closed curve,

$$\dot{\underline{X}} = \frac{\omega}{2\pi} \oint_C \frac{(\underline{x} - \underline{X})(x - \underline{X}) \cdot d\underline{x}}{|\underline{x} - \underline{X}|^2} \, .$$

The normal component of the velocity of the boundary point $\underline{X}$ is

$$\hat{N}\cdot\dot{\underline{X}} = \frac{\omega}{2\pi}\int \frac{\hat{N}\cdot(\underline{x}-\underline{X})(\underline{x}-\underline{X})\cdot d\underline{x}}{|\underline{x}-\underline{X}|^2}$$

and the contribution to $\hat{N}\cdot\dot{\underline{X}}$ at $\underline{x}=\underline{X}$ is zero.

There are a number of ways to perform the numerical integration: the one we have set up fits a periodic, cubic spline to the curve $\underline{x}(\theta)$, i.e.

$$\underline{x}(\theta) = \sum_{1}^{N} \underline{\alpha}_i B_i(\theta) + O(h^4)$$

where B_i is the cubic B-spline and h is $|\underline{x}_i - \underline{x}_{i+1}|$. The derivative

$$d\underline{x} = \sum_{1}^{N} \underline{\alpha}_i B_i'(\theta)d\theta + O(h^3 d\theta)$$

gives the tangent vector, the normal is obtained by rotation, and the normal component of $\dot{\underline{X}}$ is calculated in a consistent approximation by Simpson's rule.

With one iteration of central differences for the time integration (Huen's method) the algorithm performs well, but we are not yet satisfied with it. Further work on this subject is planned.

8. Graphics

Visualization of two and three dimensional flows creates special problems and a need for special software that isn't generally found in standard packages. Graphics functions we developed for our own purposes include:

1) Standard plotting of rough graphs at a terminal or better quality graphs on an X-Y plotter.

2) Rough representation of direction fields and contour maps at a terminal.

3) Contour maps on X-Y plotters.

4) The linear algebra (affine transformations and catenations) of two-dimensional directed arcs.

5) The linear algebra of space curves, including orthographic and stereographic projections and binocular pairs of projections.

Item 4) was of such a general utility that it has now been included in the public APL software library at Brown University. Items 3) and 5) will probably be included in the public domain, as well, but in any case, listings of any of our graphics software functions are available on request.

The most challenging graphics problem was the generation of contour maps. By comparison with other contour map algorithms we have seen, our approach is somewhat different, and it appears to be considerably more efficient. A description of the contour map algorithm is included here in appendix F.

Appendix A

Abstract

This paper contains the formulation of a numerical method to treat one, two or three dimensional unsteady flow. The method is closely related to the PIC method and finite element methods. The elements (finite fluid elements) are localized distributions of mass, characterized by mass, radius and position in space. They move according to interactions between neighboring elements that are derived from the fluid equations. Rough estimates of the dependence of the accuracy of the approximation on particle radii and the number of neighbors retained are calculated.

1. Introduction

This paper is devoted to the formulation of a numerical method to treat one, two, or three-dimensional unsteady flow of a barotropic, Stokes fluid. The generalization of the method to treat an arbitrary compressible fluid will be deferred until the present model has been implemented, and its worth as a practical algorithm, appraised.

The method to be developed here is closely related to the PIC method (Particle In Cell, Refs. 1-3), but in fact, the two essential features of that method will both be treated somewhat differently. The particles will not be treated as point-masses; instead they will be taken to be localized distributions of mass of finite extent, and as we shall see, they will be allowed to overlap. The addition of another parameter, an effective radius, for each particle allows another degree of freedom for optimization, and thus an inherently more accurate representation of a continuous density field by a finite set of localized particles. The equations of motion of the particles are derived directly from the equations of fluid mechanics, and the constitutive relations of the fluid are modelled by providing appropriate interactions between particles. Here, as in the PIC method, closure of the dynamical system of particle motions is effected by fitting mass and momentum density fields to provide a continuum approximation of the motion of the particles. The system is then closed by adopting the constitutive relations of the fluid that is to be described. Instead of the fixed cells of the PIC method, cells that move with the particles will be employed.

Mass and momentum density in the neighborhood of each particle will be defined by weighted averages, over the particle and its neighbors, of the corresponding particle attributes. This means that the implementation of the method for two- and three-dimensional flows will require the tabulation of a continually updated list of the near neighbors of each particle. It does not appear that this is a burden, however, since the major contribution to the weighted averages comes from the six or twelve nearest neighbors in two or three dimensions, respectively, and the updating can easily be accomplished by periodically checking the positions of second neighbors, which are easily assembled from the neighbors' neighbors.

The method is also closely related to the finite element methods that have been used in continuum mechanics (Refs. 4,5). However, two departures from the usual scheme of things will be introduced: In the first place; the contiguous, nonoverlapping elements, typically triangles or quadrilaterals in two dimensions and tetrahedra and various prisms in three, will be abandoned in favor of simpler elements, the finite particles, that fill space in an additive way rather than by mutual exclusion; i.e. they overlap. Thus the element parameters, nodal values, coefficients of shape functions, etc., are replaced by particle attributes and, as we shall see, cell attributes. The basic similarity between this and other finite element methods is in the determination of the particle and cell attributes by Galerkin's method where the integrated square of the error of the finite approximation is minimized.

The second departure from other finite element methods is that this one is entirely a Lagrangian formulation: the velocity of each particle is among the attributes to be defined by minimization of the error of the approximation. The velocities are then to be integrated to find the positions of the particles, thus giving a formulation that closely mimics the fluid elements on which fluid mechanics is founded.

With the overlapping of particles and the use of Galerkin's method to govern the coefficients of an additive covering of space, the present method bears at least a superficial resemblance to spectral methods that have been used in numerical fluid mechanics (Ref. 6). In the spectral methods the 'elements' are trigonometric functions and/or members of various sets of orthogonal polynomials, none of which are localized in space. The entity that corresponds to an inner product (here denoted by $\langle f_i f_j \rangle$) is the integral over all space of the product of the i^{th} and j^{th} mass distributions. The normalization is different, however; it is $\langle f_i \rangle = 1$ here. The matrix $\langle f_i f_j \rangle$ changes with time, and even the location of the largest off-diagonal elements changes in a shear flow; thus its inversion has to be done numerically. Because of the localization of the particles, however, the major off-diagonal contributions to the i^{th} row come from relatively few near neighbors of the i^{th} particle.

2. Fluid Equations

The fluid model, a barotropic, Stokes fluid includes some features of compressible flow. Specifically, the gradient of the pressure is retained in the momentum equation, but the thermodynamics is simplified by taking the pressure to be a function of density alone. The Stokes approximation defines stress in terms of a single viscosity coefficient, also a function of density, and the theory is closed without the need for an energy equation. Enough of the complexity of a compressible fluid is retained, so that the complementary functions of the particles and the cells in the finite fluid element model can be fully appreciated.

The equations that have a tensorial character will be given twice, first in the cartesian tensor notation and then in an abbreviated notation that will be used in later sections where the subscripts will denote particle and/or cell number. The cartesian tensor subscripts take on one, two or three values in as many space dimensions, summation over repeated indices is implied, and δ_{ij} is one if i=j, zero otherwise.

Conservation of mass:

$$\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x_j} (\rho u_j) = 0$$

(1)

$$\rho_t + \nabla \cdot (\rho u) = 0$$

Momentum:

$$\frac{\partial}{\partial t}(\rho u_i) + \frac{\partial}{\partial x_j}(\rho u_j u_i) = \rho g_i + \frac{\partial}{\partial x_j}\sigma_{ji}$$

(2)

$$(\rho u)_t + \nabla\cdot(\rho uu) = \rho g + \nabla\cdot\sigma$$

where g is a prescribed external force per unit mass, and the stress (with Stokes' approximation) is

$$\sigma_{ij} = \mu\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} - \frac{2}{3}\left(\frac{\partial u_k}{\partial x_k}\right)\delta_{ij}\right) - p\delta_{ij}$$

(3)

$$\sigma = 2\mu\hat{\nabla u} - \left(p + \frac{2}{3}\mu\nabla\cdot u\right)I.$$

Finally, for the barotropic model:

$$p = \bar{p}(\rho)$$

(4)

$$\mu = \bar{\mu}(\rho)$$

where the functions, $\bar{p}$ and $\bar{\mu}$, are prescribed.

3. Particle attributes

Just as in other finite element methods, the individual elements, particles in this case, can be assigned an arbitrarily complex internal structure. This leads to the usual trade-off: the accuracy of the approximation can be improved either by increasing the number of elements per unit volume or by augmenting the internal complexity of the elements. The choice is by no means a trivial one, for though the more complex elements necessitate more equations to govern the values of their several attributes, the greater separation of elements tends to allow larger time-increments in the numerical integration of the dynamical system. The question of improvement of the approximation will be deferred here, and the particles will be assigned a relatively simple internal structure.

The particle approximation consists in the replacement of the density field, $\rho(x,t)$, by a set of N localized distributions of mass, i.e.

$$(1) \qquad \rho \sim \Sigma m_i(t) f_i(x,t)$$

where the sum is from i=1 to N. The identification of m_i as the total mass of the i^{th} particle is effected by the conditions on the distribution functions,

$$(2). \qquad \langle f_i \rangle = 1 \qquad i=1,\ldots N,$$

where the angular brackets denote the integral over all space.

Since the distribution functions represent the location of mass within the particles, the location of the i^{th} particle is the mean value,

$$(3) \qquad X_i(t) = \langle x f_i \rangle,$$

and the effective radius of the i^{th} particle is defined in terms of the variance as

$$(4) \qquad r_i(t) = \langle |x-X_i|^2 f_i \rangle^{1/2} .$$

If the attributes, mass, position and effective radius, are accepted as a sufficient description of the particles, then a particularly convenient choice of distribution functions is the normal distribution,

$$(5) \qquad f_i(x,t) = \frac{1}{(\pi\sigma_i^2)^{n/2}} e^{-|x-X_i|^2/\sigma_i^2}$$

where the value of n is 1, 2 or 3 in n space dimensions, and

$$(6) \qquad \sigma_i(t) = \sqrt{\frac{2}{n}}\, r_i(t).$$

It may be noted that the first few moments, eqs.(2,3,4), are far from sufficient information to determine the distribution functions; the arbitrary and convenient choice of normal distributions relates higher moments to the radius in ways that are particularly easy to evaluate. A greater convenience, however, is the ease with which several kinds of matrix elements that will be introduced in due course can be evaluated explicitly. Thus, for example,*

* See Sec. 6, Matrix elements.

$$(7) \qquad \langle f_i f_j \rangle = \frac{1}{(\pi(\sigma_i^2+\sigma_j^2))^{n/2}} e^{-|X_i-X_j|^2/(\sigma_i^2+\sigma_j^2)} .$$

With the normal distributions, eq.(5), the mass of the i^{th} particle is distributed symmetrically about $X_i(t)$, and the approximation to the fluid momentum field that is consistent with eq.(1) is,

$$(8) \qquad \rho u \sim \Sigma m_i \dot{X}_i f_i$$

where $u(x,t)$ is the fluid velocity field. Clearly, X_i and $\dot{X}_i$ cannot be specified independently, so to preserve the character of the initial value problem for the fluid, the parameters, m_i and $\dot{X}_i$ are chosen to minimize the error of the approximations of eqs.(1 & 8). For the particle approximation the error will be taken to be the unbiased r.m.s. error,

$$(9) \qquad E \equiv \frac{1}{\rho_*} \langle (\rho - \Sigma m_i f_i)^2 + \frac{1}{U_*^2} |\rho u - \Sigma m_i \dot{X}_i f_i|^2 \rangle^{1/2} ,$$

where U_* is a characteristic fluid velocity that has been introduced to render eq.(9) dimensionally consistent, ρ_* is a characteristic fluid density, and $[E^2]$ is volume. The choice of eq.(9) is somewhat of a compromise, for though an unbiased relative error, e.g.

$$(10) \qquad \left\langle \frac{(\rho - \Sigma m_i f_i)^2}{\rho^2} + \frac{|\rho u - \Sigma m_i \dot{X}_i f_i|^2}{|\rho u|^2} \right\rangle,$$

might provide a better representation of the fields, the theory that follows therefrom is much more difficult.

The conditions to minimize E are:

$$(11) \qquad \rho_*^2 E \frac{\partial E}{\partial m_i} = \langle \rho f_i \rangle - \Sigma m_j \langle f_i f_j \rangle + \frac{\rho^2 E}{m_i} \dot{X}_i \cdot \frac{\partial E}{\partial \dot{X}_i} = 0$$

$$(12) \qquad \frac{\rho_*^2 U_*^2 E}{m_i} \frac{\partial E}{\partial \dot{X}_i} = \langle \rho u f_i \rangle - \Sigma m_j \dot{X}_j \langle f_i f_j \rangle = 0$$

$$(13) \qquad \frac{\rho_*^2 E}{m_i} \frac{\partial E}{\partial \sigma_i} = \langle f_{\sigma_i} (\rho - \Sigma m_j f_j) \rangle + \frac{1}{U_*^2} \dot{X}_i \cdot \langle f_{\sigma_i} (\rho u - \Sigma m_j \dot{X}_j f_j) \rangle = 0$$

where

$$(14) \qquad f_{\sigma_i} \equiv \frac{\partial f_i}{\partial \sigma_i} = \frac{2}{\sigma_i^3} \left(|x - X_i|^2 - \frac{n\sigma_i^2}{2} \right) f_i \ .$$

Note that eq.(12) implies the vanishing of the corresponding term of eq.(11), and eqs.(11 and 12) allow the replacement of f_{σ_i} by $|x-X_i|^2 f_i$ in eq.(13).

Now it may be noted that if the conditions of equation (13) are retained, the result is a theory in which the σ's are large, the 'particles' are not localized, and near and far neighbors are of equal importance in equations (11) and (12). Accordingly, equation (13) will be dropped altogether, and in §7 estimates will be given for choices of σ's that minimize the error of truncated approximations.

4. Particle dynamics

Consider first eqs.(3.11,3.12):

(1) $$\Sigma m_j \langle f_i f_j \rangle = \langle \rho f_i \rangle$$

(2) $$\Sigma m_j \dot{X}_j \langle f_i f_j \rangle = \langle \rho u f_i \rangle \ .$$

In the description of unsteady flow, the parameters, m_i, $\dot{X}_i$ and σ_i, are to vary with time, but in such manner that eqs.(1,2) are preserved. Equations (1,2) can thus be regarded as constraints on the particle dynamics; to see their effects, it suffices to differentiate them to obtain

(3) $$(\Sigma m_j \langle f_i f_j \rangle)^{\cdot} = \langle \rho f_i \rangle^{\cdot} = \langle \rho \dot{f}_i + f_i \rho_t \rangle$$

(4) $$(\Sigma m_j \dot{X}_j \langle f_i f_j \rangle)^{\cdot} = \langle \rho u f_i \rangle^{\cdot} = \langle \rho u \dot{f}_i + f_i (\rho u)_t \rangle \ .$$

The subscript, i, runs from 1 to N in eqs.(3,4), which can be thought of as the mass and momentum transport laws of the particle dynamics. The right-hand sides of eqs.(3,4) contain both particle attributes and the fluid density and momentum fields; they are the particle interactions that are to be determined presently by the introduction of cells.

From the fluid eqs.(2.1,2.2) and eq.(3.14) it follows that

(5) $$\langle \rho f_i \rangle^{\cdot} = \frac{2\dot{\sigma}_i}{\sigma_i^3} \langle \rho (|x - X_i|^2 - \frac{n\sigma_i^2}{2}) f_i \rangle$$

$$- \langle \rho \nabla f_i \rangle \cdot \dot{X}_i - \langle f_i \nabla \cdot \rho u \rangle$$

$$(6)\qquad \langle \rho u f_i \rangle^{\cdot} = \frac{2\dot{\sigma}_i}{\sigma_i^3} \langle \rho u (|x-X_i|^2 - \frac{n\sigma_i^2}{2}) f_i \rangle$$

$$- \langle \rho u \nabla f_i \rangle \cdot \dot{X}_i - \langle f_i \nabla \cdot \rho u u \rangle + \langle f_i (\rho g + \nabla \cdot \sigma) \rangle$$

Clearly, eqs.(3,4,5,6) are incomplete, even if the mass and momentum densities were known; there are only 2N relations for the time derivatives of 3N particle attributes. The remaining N equations can be obtained by differentiating the estimates of $\sigma_i(t)$ that will be given in §7, or eqs. (3,4,5,6) can be iterated with $\dot{\sigma}_i = 0$ in the leading approximation.

Some estimates of the relative importance of various terms in eqs.(5,6) can be made as follows: Let it be supposed that the Taylor series of the mass and momentum densities were known, i.e.

$$(7)\qquad \rho = \rho_i + (x-X_i)\cdot(\nabla\rho)_i + \tfrac{1}{2}(x-X_i)(x-X_i):(\nabla\nabla\rho)_i + \ldots$$

$$(8)\qquad \rho u = (\rho u)_i + (x-X_i)\cdot(\nabla\rho u)_i + \tfrac{1}{2}(x-X_i)(x-X_i):(\nabla\nabla\rho u)_i + \ldots$$

where the subscripted quantities are evaluated at $x = X_i$. The substitution of the i^{th} Taylor series in the i^{th} of eqs.(5,6), an integration by parts of the matrix elements that contain ∇f_i, and the results for normal distributions,*

$$(9)\qquad \langle |x-X_i|^2 f_i \rangle = \frac{n\sigma_i^2}{2}$$

* See Sec. 6, Matrix elements.

$$(10) \qquad \langle |x-X_i|^4 f_i \rangle = (\frac{n}{2} + \frac{n^2}{4})\sigma_i^4 \ ,$$

give the results,

$$(11) \qquad \langle \rho f_i \rangle^{\cdot} = (\frac{1}{2}\ \sigma_i \dot{\sigma}_i (\Delta\rho)_i + (\nabla\rho)_i \cdot \dot{X}_i - (\nabla\cdot\rho u)_i)(1+0(\sigma_i^2))$$

$$(12) \qquad \langle \rho u f_i \rangle^{\cdot} = (\frac{1}{2}\ \sigma_i \dot{\sigma}_i (\Delta\rho u)_i + (\nabla\rho u)_i \cdot \dot{X}_i - (\nabla\cdot\rho uu)_i$$

$$+\rho_i g_i + (\nabla\cdot\sigma)_i)(1+0(\sigma_i^2)).$$

The coefficients of the relative rates of change $(\dot{\sigma}_i/\sigma_i)$ are of the order, $0(\sigma_i^2)$, and thus as the number of particles per unit volume is increased, the error introduced by neglecting those terms of eqs.(11,12) vanishes as

$$(13) \qquad \sigma_i^2 = 0((V/N)^{2/n}) \ \rightarrow 0 \ .$$

In the same limit, u_i approaches $\dot{X}_i$ as N/V becomes large, and the major contributions to the particle dynamics are:

$$(14) \qquad (\Sigma m_j \langle f_i f_j \rangle)^{\cdot} + \rho_i (\nabla\cdot u)_i \approx 0$$

$$(15) \qquad (\Sigma m_j \dot{X}_j \langle f_i f_j \rangle)^{\cdot} + (\rho u)_i (\nabla\cdot u)_i \approx \rho_i g_i + (\nabla\cdot\sigma)_i$$

where, once again, the subscripts on the fields and their gradients indicate evaluation at $x = X_i(t)$.

5. Cells

From the results of the previous section it can be seen that the least information about the fluid mass and momentum densities that is needed to complete the particle dynamics approximately includes the values at $x = X_i$, ρ_i and $(\rho u)_i$, the first derivatives, $(\nabla\rho)_i$ and $(\nabla\rho u)_i$, for the evaluation of $(\nabla\cdot u)_i$, and the second derivatives, $(\nabla\nabla\rho)_i$ and $(\nabla\nabla\rho u)_i$, for the evaluation of $(\nabla\cdot\sigma)_i$. For that purpose then, cells are introduced to provide local polynomial approximations where

$$(1) \qquad \rho \sim \rho_i + (x-X_i)\cdot(\nabla\rho)_i + \tfrac{1}{2}(x-X_i)(x-X_i):(\nabla\nabla\rho)_i$$

$$(2) \qquad \rho u \sim (\rho u)_i+(x-X_i)\cdot(\nabla\rho u)_i + \tfrac{1}{2}(x-X_i)(x-X_i):(\nabla\nabla\rho u)_i \ .$$

The subscripted quantities in eqs.(1,2) are not the fields and their derivatives evaluated at $x = X_i$, as in the previous section; rather, they are the coefficients of quadratic approximations of mass and momentum density fields that would be obtained if the system of particles were treated, after the manner of the kinetic theory of gases, in a continuum approximation. The process is the reverse of kinetic theory, however, since it is the particle interactions that are unknown here, and the mass and momentum densities are to be found in order to adjust the particle interactions to suit the thermodynamic properties of the fluid at hand.

The local quadratic approximations for the mass and momentum densities are determined by minimization of the biased r.m.s. error,

$$(3) \qquad E_i \equiv \frac{1}{\rho_*} \langle F_i[(\rho_i+(x-X_i)\cdot(\nabla\rho)_i+ \tfrac{1}{2}(x-X_i)(x-X_i):(\nabla\nabla\rho)_i-\Sigma m_j f_j)^2+$$

$$+ \frac{1}{U_*^2} ((\rho u)_i+(x-X_i)\cdot(\nabla\rho u)_i+ \tfrac{1}{2}(x-X_i)(x-X_i):(\nabla\nabla\rho u)_i$$

$$- \Sigma m_j \dot{X}_j f_j)^2]\rangle^{1/2}$$

where $F_i(x,t)$ is a localized weight function that is normalized,

$$(4) \qquad \langle F_i \rangle = 1,$$

located at $X_i(t)$,

$$(5) \qquad X_i(t) = \langle xF_i \rangle \ ,$$

and has an effective radius,

$$(6) \qquad R_i(t) = \langle |x-X_i|^2 F_i \rangle^{1/2}.$$

Again, for the ease with which matrix elements can be evaluated explicitly, the weight functions are taken to be the normal distributions,

$$(7) \qquad F_i(x,t) = \frac{1}{(\pi\Sigma_i^2)^{n/2}} e^{-|x-X_i|^2/\Sigma_i^2}$$

where n is the number of space dimensions and

$$(8) \qquad \Sigma_i(t) = \sqrt{\frac{2}{n}}\, R_i(t).$$

This time, the quantities that enter in the conditions to minimize E_i are the subscripted field quantities and Σ_i; the conditions are:

$$(9) \qquad \rho_i + \frac{1}{2} \langle (x-X_i)(x-X_i)F_i \rangle :(\nabla\nabla\rho)_i = \Sigma m_j \langle F_i f_j \rangle$$

$$(10)\qquad (\rho u)_i + \frac{1}{2}\langle (x-X_i)(x-X_i)F_i\rangle : (\nabla\nabla\rho u)_i = \Sigma m_j \dot{X}_j \langle F_i f_j\rangle$$

$$(11)\qquad \langle (x-X_i)(x-X_i)F_i\rangle \cdot (\nabla\rho)_i = \Sigma m_j \langle (x-X_i)F_i f_j\rangle$$

$$(12)\qquad \langle (x-X_i)(x-X_i)F_i\rangle \cdot (\nabla\rho u)_i = \Sigma m_j \dot{X}_j \langle (x-X_i)F_i f_j\rangle$$

$$(13)\qquad \langle (x-X_i)(x-X_i)F_i\rangle \rho_i + \frac{1}{2}\langle (x-X_i)(x-X_i)(x-X_i)(x-X_i)F_i\rangle : (\nabla\nabla\rho)_i = \Sigma m_j \langle (x-X_i)(x-X_i)F_i f_j\rangle$$

$$(14)\qquad \langle (x-X_i)(x-X_i)F_i\rangle (\rho u)_i + \frac{1}{2}\langle (x-X_i)(x-X_i)(x-X_i)(x-X_i)F_i\rangle : (\nabla\nabla\rho u)_i = \Sigma m_j \dot{X}_j \langle (x-X_i)(x-X_i)F_i f_j\rangle$$

and

$$(15)\qquad \langle F_{\Sigma_i}[(\rho_i + (x-X_i)\cdot(\nabla\rho)_i + \frac{1}{2}(x-X_i)(x-X_i):(\nabla\nabla\rho)_i - \Sigma m_j f_j)^2 + \frac{1}{U_*^2}((\rho u)_i + (x-X_i)\cdot(\nabla\rho u)_i + \frac{1}{2}(x-X_i)(x-X_i):(\nabla\nabla\rho u)_i - \Sigma m_j \dot{X}_j f_i)^2]\rangle = 0$$

where

$$(16)\qquad F_{\Sigma_i} \equiv \frac{\partial F_i}{\partial \Sigma_i} = \frac{2}{\Sigma_i^3}\left(|x-X_i|^2 - \frac{n\Sigma_i^2}{2}\right)F_i .$$

In eqs.(9 to 14) the terms containing odd moments of F_i have already been dropped; in the next section, further simplifications that follow from the choice of normal distributions will be derived.

The final condition, eq.(15) which governs the choice of cell radii to minimize the error of the approximation, implies the inappropriate result,

$\Sigma_i = 0$. Accordingly, it will be dropped, along with the corresponding condition of eq.(3.13), and other estimates of both σ_i and Σ_i will be given in the section on particle and cell radii.

Given the quadratic approximations for the mass and momentum densities, the corresponding fluid velocity is

$$(17) \quad u = u_i+(x-X_i)\cdot(\nabla u)_i + \tfrac{1}{2}(x-X_i)(x-X_i):(\nabla\nabla u)_i + O(|x-X_i|^3)$$

where

$$(18) \quad u_i = (\rho u)_i/\rho_i$$

$$(19) \quad (\nabla u)_i = ((\nabla\rho u)_i - u_i(\nabla\rho)_i)/\rho_i$$

and

$$(20) \quad (\nabla\nabla u)_i = ((\nabla\nabla\rho u)_i-u_i(\nabla\nabla\rho)_i-(\nabla u)_i(\nabla\rho)_i-(\nabla\rho)_i(\nabla u)_i)/\rho_i.$$

The thermodynamic quantities that appear in $\nabla\cdot\sigma$ are

$$(21) \quad p \sim \bar{p}(\rho_i)+\bar{p}'(\rho_i)(x-X_i)\cdot(\nabla\rho)_i + \\ + \tfrac{1}{2}(x-X_i)(x-X_i):(\bar{p}'(\rho_i)(\nabla\nabla\rho)_i+\bar{p}''(\rho_i)(\nabla\rho)_i(\nabla\rho)_i)$$

and

$$(22) \quad \mu \sim \bar{\mu}(\rho_i) + \bar{\mu}'(\rho_i)(x-X_i)\cdot(\nabla\rho)_i + \\ + \tfrac{1}{2}(x-X_i)(x-X_i):(\bar{\mu}'(\rho_i)(\nabla\nabla\rho)_i+\bar{\mu}''(\rho_i)(\nabla\rho)_i(\nabla\rho)_i).$$

6. Matrix elements

The choice of the normal distributions,

$$(1) \quad f_i = \frac{1}{(\pi\sigma_i^2)^{n/2}} e^{-|x-X_i|^2/\sigma_i^2}$$

$$(2) \quad F_i = \frac{1}{(\pi\Sigma_i^2)^{n/2}} e^{-|x-X_i|^2/\Sigma_i^2} ,$$

leads to numerous simplifications of the several matrix elements that have been introduced. Generally speaking, the simplification of the tensorial character of the matrix elements is a consequence of the choice of functions of $|x-X_i|^2$, and the relative ease with which they can be evaluated is a consequence of the specific choice.

Consider first the right-hand side of eqs.(5.9 to 5.14): The integral $\langle F_i f_j \rangle$ is of the same form as

$$(3) \quad \langle f_i f_j \rangle = \frac{1}{(\pi^2\sigma_i^2\sigma_j^2)^{n/2}} \int e^{-|x-X_i|^2/\sigma_i^2} e^{-|x-X_j|^2/\sigma_j^2} dx$$

and the integral $\langle F_i f_j f_k \rangle$ that will appear in the next section. The integral of the product of any number of normal distributions can be evaluated by introducing the 'center of mass',

$$(4) \quad X = \left(\Sigma \frac{X_i}{\sigma_i^2}\right) / \Sigma \frac{1}{\sigma_i^2} .$$

The exponent in the integrand is then

$$(5) \quad \Sigma \frac{|x-X_i|^2}{\sigma_i^2} = |x-X|^2 \Sigma \frac{1}{\sigma_i^2} + \Sigma \frac{|X-X_i|^2}{\sigma_i^2} + 2(x-X)\cdot\Sigma \frac{(X-X_i)}{\sigma_i^2} .$$

The last term vanishes by the choice of X, the second term gives the dependence on the coordinates X_i, and the integral of the first

term gives the normalization. The result for the product of two can be rearranged to give eq.(3.7) and

$$(6)\quad \langle F_i f_j \rangle = \frac{1}{(\pi(\Sigma_i^2+\sigma_j^2))^{n/2}}\, e^{-|X_i-X_j|^2/(\Sigma_i^2+\sigma_j^2)} .$$

Likewise the result for the product of three normal distributions can be rearranged in the form

$$(7)\quad \langle F_i f_j f_k \rangle = \frac{1}{(\pi^2(\Sigma_i^2(\sigma_j^2+\sigma_k^2)+\sigma_j^2\sigma_k^2))^{n/2}}$$

$$e^{-\left(\frac{\Sigma_i^2|X_j-X_k|^2+\sigma_j^2|X_k-X_i|^2+\sigma_k^2|X_j-X_i|^2}{\Sigma_i^2(\sigma_j^2+\sigma_k^2)+\sigma_j^2\sigma_k^2}\right)}.$$

The rearrangement of the general result is somewhat more tedious than the direct evaluation of eq.(7) by completion of the square in the exponent.

Next, in order of appearance, is

$$(8)\quad \langle (x-X_i)F_i f_j \rangle = \frac{\Sigma_i^2}{2} \langle F_i f_j \rangle X_i$$

$$= \frac{\Sigma_i^2}{\Sigma_i^2+\sigma_j^2} (X_j-X_i) \langle F_i f_j \rangle ,$$

and then

$$(9)\quad \langle (x-X_i)(x-X_i)F_i f_j \rangle = \frac{\Sigma_i^2}{2} (\langle (x-X_i)F_i f_j \rangle X_i + I\langle F_i f_j \rangle)$$

$$= \left(\frac{\Sigma_i^4}{(\Sigma_i^2+\sigma_j^2)^2} (X_j-X_i)(X_j-X_i) + \right.$$

$$\left. + \frac{\Sigma_i^2\sigma_j^2}{2(\Sigma_i^2+\sigma_j^2)} I\right) \langle F_i f_j \rangle$$

where I is the unit dyad (equal δ_{ij} in cartesian tensor notation).

On the left-hand sides of eqs.(5.9 to 5.14) the matrix elements are:

$$(10) \quad \langle (x-X_i)(x-X_i)F_i \rangle = \frac{\Sigma_i^2}{2} I$$

and

$$(11) \quad \langle (x-X_i)(x-X_i)(x-X_i)(x-X_i)F_i \rangle = \frac{1}{4} \Sigma_i^4 S$$

where S is the symmetric sum of outer products of unit dyads that has the cartesian tensor representation,

$$(12) \quad S_{ijk\ell} = \delta_{ij}\delta_{k\ell} + \delta_{ik}\delta_{j\ell} + \delta_{i\ell}\delta_{jk} .$$

The results (10,11,12) follow by the substitutions of $(x-X_i)$ and $(x-X_i)(x-X_i)(x-X_i)$ for G(x) in the identity,

$$(13) \quad \langle (x-X_i)F_iG \rangle = \frac{-\Sigma_i^2}{2} \langle \nabla F_i G \rangle = \frac{\Sigma_i^2}{2} \langle F_i \nabla G \rangle .$$

Contractions of (10,11,12) then give the results (4.9,4.10).

This completes the evaluation of matrix elements, except for moments of $\langle F_i f_j f_k \rangle$ that can be evaluated after the manner employed for eqs.(8,9). With the simplification that follows from the choice of normal distributions, eqs.(5.9-5.14) now are:

$$(14) \quad \rho_i + \frac{1}{4} \Sigma_i^2 (\Delta\rho)_i = \Sigma m_j \langle F_i f_j \rangle$$

$$(15) \quad (\nabla\rho)_i = 2\Sigma m_j \frac{X_j - X_i}{\Sigma_i^2 + \sigma_j^2} \langle F_i f_j \rangle$$

$$(16)\qquad \rho_i I + \frac{1}{4}\Sigma_i^2((\Delta\rho)_i I + 2(\nabla\nabla\rho)_i) =$$

$$= 2\Sigma m_j\left(\frac{\Sigma_i^2(X_j-X_i)(X_j-X_i)}{(\Sigma_i^2+\sigma_j^2)^2} + \frac{\sigma_j^2 I}{2(\Sigma_i^2+\sigma_j^2)}\right) \langle F_i f_j\rangle \ .$$

From eq.(14) and the trace of eq.(16) it follows that

$$(17)\qquad (\Delta\rho)_i = 4\Sigma \frac{m_j}{\Sigma_i^2+\sigma_j^2}\left(\frac{|X_j-X_i|^2}{\Sigma_i^2+\sigma_j^2} - \frac{n}{2}\right) \langle F_i f_j\rangle$$

$$(18)\qquad \rho_i = \Sigma m_j\left(1 - \frac{\Sigma_i^2}{\Sigma_i^2+\sigma_j^2}\left(\frac{|X_j-X_i|^2}{\Sigma_i^2+\sigma_j^2} - \frac{n}{2}\right)\right) \langle F_i f_j\rangle \ .$$

The corresponding relations between coefficients for ρu are identical except that m_j is everywhere replaced by $m_j\dot{X}_j$.

7. Particle and cell radii

At the end of §4 it was mentioned that the attempt to minimize the unbiased error by choice of σ's leads to a nonlocalized theory in which the 'particle' radii are large. The result can be seen by considering the 1-dimensional, uniform, periodic case where $\rho=1$, $-\infty < x < \infty$, the particles are equally spaced at $X_i = i$ with equal masses $m_i = m$, and $\sigma_i = \sigma$. Then

$$E^2 = \langle(1-m\Sigma f_i)^2\rangle \tag{1}$$

and the relation between m and σ is

$$\langle f_i\rangle = 1 = m\Sigma\langle f_i f_j\rangle = \frac{m}{\sqrt{2\pi}\sigma} \sum_{-\infty}^{\infty} e^{-k^2/2\sigma^2} . \tag{2}$$

From equations (1,2) it follows (as in the derivation of Bessel's inequality for orthogonal functions) that

$$\begin{aligned} E^2 &= \langle 1\rangle - 2m\,\Sigma\langle f_i\rangle + m^2\,\Sigma\Sigma\,\langle f_i f_j\rangle \\ &= \langle 1\rangle - m^2\,\Sigma\Sigma\langle f_i f_j\rangle \\ &= \langle 1-m\rangle . \end{aligned} \tag{3}$$

Since the particles are equally spaced the error per particle is

$$e^2 = 1-m = 1 - \sqrt{2\pi}\,\sigma/\sum_{-\infty}^{\infty} e^{-k^2/2\sigma^2} . \tag{4}$$

Figure 1 is a graph of e^2 for $0 < \sigma \leq 1$, showing its decline to zero as $\sigma \to \infty$: the notable feature is the rapid approach of e^2 to negligibly small values. A similar phenomenon takes place in two and three dimensions.

Given that σ's will be relatively small, the next question is the truncation error associated with the approximate evaluation of m's from

$$(5) \qquad \langle \rho f_i \rangle \approx \sum_{\leq N} m_j \langle f_i f_j \rangle$$

where the sum is over j=i and the N nearest neighbors of the ith particle. In the uniform periodic case the m's and σ's are equal, ρ is 1, and the particles are at the integers in one dimension, in a hexagonal lattice in two, and in one of the close-packed lattices in three. In any case

$$(6) \qquad 1 = m_N(\sigma) \sum_{\leq N} \langle f_o f_j \rangle \ ,$$

$$(7) \qquad E^2 = \langle 1 \rangle - 2m_N \sum_{\infty} \langle f_i \rangle + m_N^2 \sum_{\infty} \sum_{\infty} \langle f_i f_j \rangle$$

$$= \langle 1 \rangle - \sum_{\infty} m_N + \sum_{\infty} \sum_{>N} m_N^2 \langle f_o f_j \rangle ,$$

and the error per particle is

$$(8) \qquad e^2 = \tau_n - m_N(1 - m_N \sum_{>N} \langle f_o f_j \rangle)$$

where τ_n is the volume per particle in the n-dimensional, close packed lattice, i.e.

$$(9) \qquad \tau_n = d, \ \frac{\sqrt{3}}{2} d^2, \ \frac{1}{\sqrt{2}} d^3 \qquad \text{for} \quad n=1,2,3$$

where d is the separation of nearest neighbors. Figures 2,3 and 4 are graphs of e^2 for truncations that include first, second and third neighbors, for various ranges of σ. The separation of nearest neighbors is set equal to one in all cases, and the three-dimensional, close-packed lattice is face-centered cubic.

A different measure of the accuracy of the truncated approximations is given in Figures 5, 6 and 7, which are graphs of $\tau_n - M_N$. The value of σ at which $M_N = \tau_n$ is the one for which mass is conserved, i.e. $\langle\rho\rangle = \Sigma m_j$. It should be noted, however, that the value of σ at which mass is conserved is systematically less than that where e^2 is minimized. By direct calculation, the values of σ for mass conservation with the nearest two, six and twelve neighbors included are:

$$(10) \qquad \frac{\sigma_n}{d} \approx 0.572,\ 0.498,\ 0.444 \quad \text{for} \quad n=1,2,3.$$

The corresponding errors per unit volume are:

$$(11) \qquad \frac{e_n^2}{\tau_n} d^n \approx 0.000311,\ 0.00887,\ 0.0258.$$

In the case where ρ is not constant and the particles are not in a periodic array, a simple estimate of $\sigma_i(t)$ is equation (10) with

$$(12) \qquad d_i(t) - \sum_{\leq N} |X_j - X_i|/N$$

where the sum is over nearest neighbors. Many other ways to estimate σ_i can be found, however, and the adoption of (12) may be considered as provisional. In any case, as the number of particles per unit volume becomes large, $\sigma_i \to 0$ along with d_i, and if ρ is differentiable, then equation (11) provides an estimate of a local relative error

$$(13) \qquad \left\langle \frac{(\rho - \Sigma m_j f_j)^2}{\rho^2} \right\rangle_i / \langle 1 \rangle_i$$

where $\langle 1 \rangle_i$ is the volume of a neighborhood of X_i that has a diameter that is large, compared to d_i, and is small, compared to the scale length $\rho/|\underline{\nabla}\rho|$.

There remains now the question of the error in the cells. The uniform, periodic case gives rise to double sums that are relatively easy to evaluate in one dimension, rather more difficult in two and three. The cell error is

$$(14) \qquad E_o^2 = \langle F_o(\rho_N - m\Sigma f_j)^2 \rangle$$

where

$$(15) \qquad \rho_N = m \sum_{\leq N} \Sigma \langle F_o f_j \rangle = \frac{m}{(\pi(\Sigma^2+\sigma^2))^{n/2}} \sum_{\leq N} e^{-|X_j|^2/(\Sigma^2+\sigma^2)}$$

and again the sum is over the N nearest neighbors. From (14, 15) it follows that

$$(16) \qquad E_o^2 = m^2 \sum_{\infty} \sum_{\infty} \Sigma \langle F_o f_i f_j \rangle - \rho_N(\rho_N + 2m \sum_{>N} \langle F_o f_j \rangle),$$

and in the one dimensional case

$$(17) \qquad \langle F_o f_i f_j \rangle = \frac{1}{\pi\sigma\sqrt{2\Sigma^2+\sigma^2}} e^{-(i-j)^2/2\sigma^2} e^{-(i+j)^2/2(2\Sigma^2+\sigma^2)}.$$

Figures 8 and 9 are graphs of $E_o^2(\Sigma)$ for various truncated approximations. The two left-arguments of the function CLERR set the number of neighbors retained in the particle and cell truncations, respectively. The particle mass is <u>one</u> in all cases, and σ is set equal to the value that conserves mass in the truncated particle approximation. The appearance of phenomenal accuracy as $\Sigma \to 0$ is to be disregarded since it refers to approximate

calculation of density within the particle at X=0. Figure 8 approximates the untruncated cell over the range $0 < \Sigma \leq 1$, and indicates that cell error is limited by the error in the particle approximation. Figure 9 shows the effect of truncation of the cell approximation.

The final figure shows $1-\rho_N$ (with m=1) in various approximations. The indication is that mass conservation provides a reasonable estimate of a value of Σ for which truncation error is relatively small and the density is not the density within a particle. In the case where the number of neighbors retained is the same for particles and cells, the estimate is

$$\Sigma_i = \sigma_i. \tag{18}$$

References

[1] Harlow, F.H.,"The particle-in-cell method for numerical solution of problems in fluid dynamics", Proceedings of Symposia in Applied Mathematics (American Mathematical Society) 15 (1963) 269-288.

[2] Harlow, F.H.,"The particle-in-cell computing method for fluid dynamics", Methods in Computational Physics (Academic Press) 3 (1964) 319-343.

[3] Dyachenko, V.F., "The free-point method for problems of continuous media", Computer Meths. in Appl. Mech. & Engrg. 2 (1973) 265-277.

[4] Nickell, R.E., Tanner, R.I. and Caswell, B., "The solution of viscous incompressible jet and free surface flows using finite-element methods", J. Fluid Mech. 65 (1974) 189-206.

[5] Finlayson, B.A., "Weighted residual methods and their relation to finite element methods in flow problems", Finite Elements in Fluids, vol. 1, John Wiley & Sons 1975.

[6] Orszag, S.A., "Numerical simulation of incompressible flows within simple boundaries. Galerkin (spectral) representations", Studies in Appl. Math. 50 (1971) 293-327, reprinted, Computer Fluid Dynamics - Recent Advances, AIAA Selected Reprint Series 15 (1973) 94-111.

```
      ⎕CR'PLOT'
GR←R PLOT Y;N
⍝  PLOTS Y0,Y1,...YN : ⍴GR IS (R+1),N+1
'RANGE: ' ,(⍕⌊/Y), ' ≤ Y ≤ ' ,⍕⌈/Y
GR←((R+1),⍴Y)⍴'|',(-1-⍴Y)⍴' '
⍝  CREATES BLANK GRAPH AND Y-AXIS
Y←((0,Y)-⌊/Y)÷(⌈/Y)-⌊/Y
⍝  CONVERTS RANGE TO [0,1] AND LOCATES Y=0
Y←⌈0.5+R×Y
⍝  CONVERTS RANGE TO [1,R+1] , INTEGERS
→((1>Y[1])∨Y[1]>R+1)/B1
GR[R+2-Y[1];]←'+',(-2-⍴Y)⍴'-'
⍝  ADDS X-AXIS IF APPROPRIATE
B1:Y←1+Y,0⍴N←-1-⍴Y
⍝  DROPS Y=0 AND SETS INITIAL INDEX FOR
B2:→(0<N←N-1)/B2,0⍴GR[R+2-Y[N];N]←'∘'
⍝  NOTE USE OF 0⍴ ... IN [B1] AND [B2]

      ⎕CR'ERROR'
Z←ERROR S;K;DZ
⍝  1 DIMENSION - NO TRUNCATION
Z←(⍴S)⍴1,0⍴S←2×S×S,0⍴K←1
BR1:→(∨/DZ>Z×1E¯10)/BR1,0⍴K←K+1,0⍴Z←Z+DZ←2×*-(K×K)÷S
Z←1-((○S)*0.5)÷Z

      19 PLOT_ERROR S←(⍳50)÷50
RANGE: 5.351E¯9 ≤ Y ≤ 0.9499
∘
|∘
|  ∘
|   ∘
|    ∘
|     ∘
|      ∘
|       ∘
|        ∘
|         ∘
|          ∘
|           ∘
|            ∘
|             ∘
|              ∘
|               ∘∘
|                 ∘∘
|                   ∘∘∘
+--------------------∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘

      ⍝  FIGURE 1
```

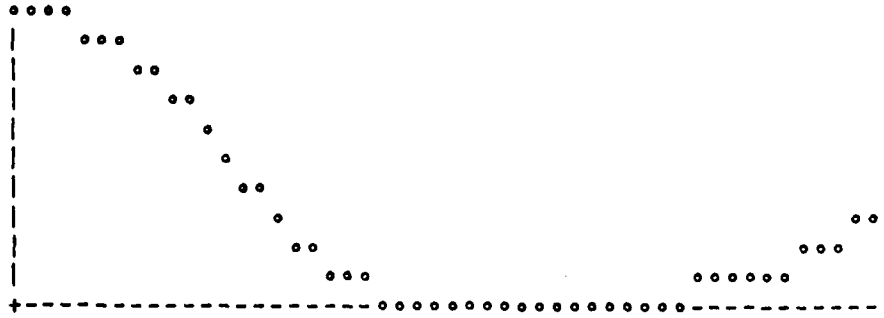
```
      ]CR 'ERR1'
Z←N ERR1 S;M;DZ
⍝  TRUNCATE AT NTH NBRS
M←((○S)*0.5)÷1+2×+/[1]*-((⍳N)×⍳N)∘.÷S←2×S×S
Z←(⍴S)⍴0,0⍴N←N+1
BR1:→(∨/DZ>2×1E¯8)/BR1,0⍴N←N+1,0⍴Z←Z+DZ←2×*-(N×N)÷S
Z←1-M×1-M×Z÷(○S)*0.5

      10 PLOT 1 ERR1 1.5×S←(⍳50)÷50
RANGE: 0.0004798 ≤ Y ≤ 0.9248
```

```
      10 PLOT 2 ERR1 2×S
RANGE: 4.438E¯6 ≤ Y ≤ 0.8997
```

```
      10 PLOT 3 ERR1 3×S
RANGE: 4.539E¯8 ≤ Y ≤ 0.8496
```

```
      ⍝  FIGURE 2
```

```
      ]CR 'ERR2'
Z←N ERR2 S;NBR;M;DZ
⍝  HEX - NTH NBRS UP TO N=10 (12 AT R=21*÷2)
⍝  ASYMPTOTIC PARTICLE DENSITY 2÷3*÷2 FOR R>21*÷2
NBR← 1 3 4 7 9 12 13 16 19 21
NBR← 2 10 ⍴NBR, 6 6 6 12 6 6 12 6 12 12
M←(○S)÷1+(N↑NBR[2;])+.××-(N↑NBR[1;])∘.÷S←2×S×S
→(N=10)/EX1
→BR1,0⍴N←22,0⍴Z←(N↑NBR[2;])+.××-(N↑NBR[1;])∘.÷S
EX1:Z←(⍴S)⍴0,0⍴N←22
BR1:→(∨/DZ>Z×1E¯8)/BR1,0⍴N←N+1,0⍴Z←Z+DZ←(○2÷3*0.5)××-N÷S
Z←(0.5×3*0.5)-M×1-M×Z÷○S

      10 PLOT 1 ERR2 S←(⍳50)÷50
RANGE: 0.001881 ≤ Y ≤ 0.8635
```

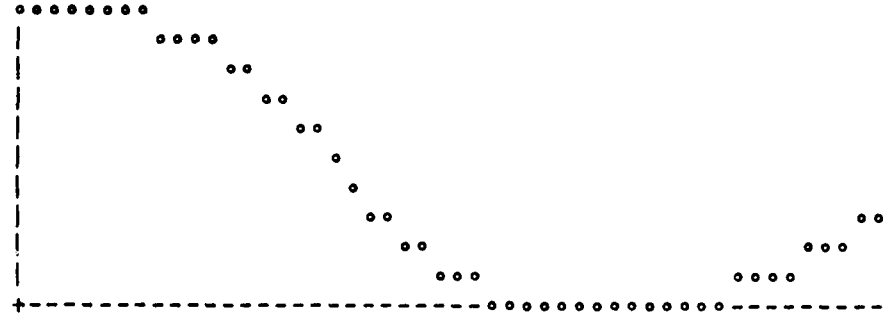


```
      8 PLOT 2 ERR2 1.2×S
RANGE: 0.0003351 ≤ Y ≤ 0.8624
```

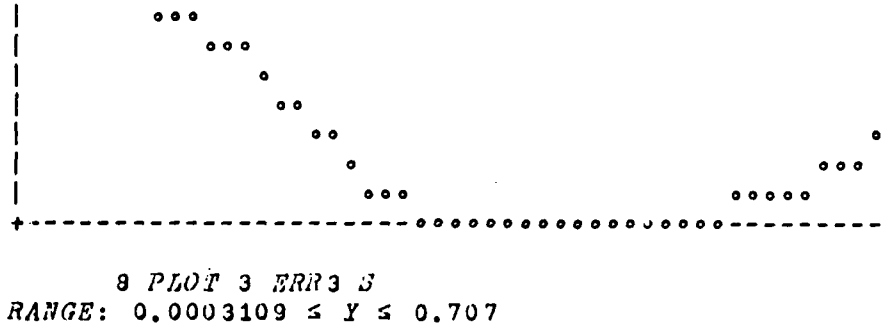
```
      8 PLOT 3 ERR2 1.5×S
RANGE: 0.00002033 ≤ Y ≤ 0.8604
```

```
      ⍝ FIGURE 3
```

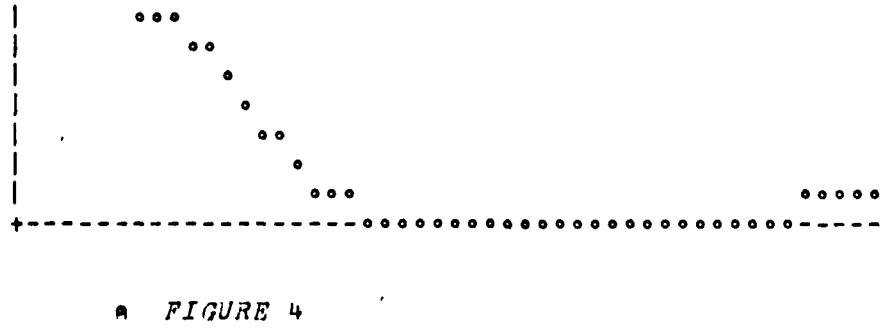
```
      ]CR 'ERR3'
Z←N ERR3 S;NBR;M1;DZ
⍝  FCC - NTH NBRS UP TO N=24 (R=5 - 3 WAYS)
⍝  ASYMPTOTIC PARTICLE DENSITY 2*÷2 FOR R>5
NBR← 12 6 24 12 24 8 48 6 36 24 24 24
NBR←NBR, 72 0 48 12 48 30 72 24 48 24 48 8 84
N←N+N>13
M1←((○S)*1.5)÷1+(N∘.×NBR)+.××-(⍳N)∘.÷S←2×S×S
→(N=25)/EX1
→BR1,0⍴N←26,0⍴Z←(N∘.×NBR)+.××-(N+⍳25)∘.÷S
EX1:Z←(⍴S)⍴0,0⍴N←26
BR1:→((∨/DZ>Z×1E¯8),0⍴N←N+1)/BR1,0⍴Z←Z+DZ←(○2×2*0.5)×(N*0.5)××-N÷S
Z←(÷2*0.5)-M1×1-M1×Z÷(○S)*1.5

      10 PLOT 1 ERR3 0.75×S←(⍳50)÷50
RANGE: 0.006815 ≤ Y ≤ 0.7071
```



```
      8 PLOT 2 ERR3 0.85×S
RANGE: 0.002623 ≤ Y ≤ 0.707
```



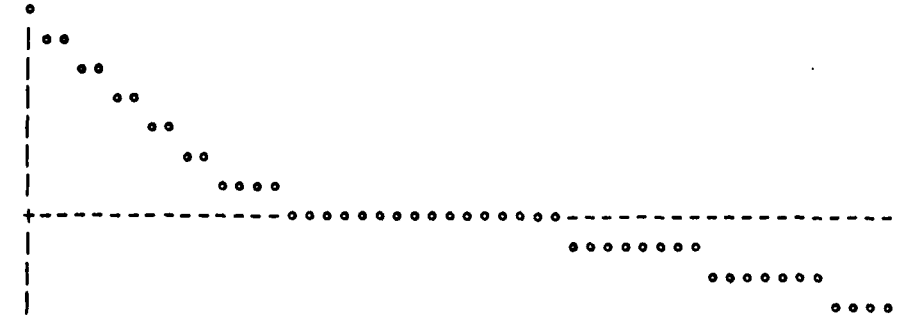
```
      8 PLOT 3 ERR3 S
RANGE: 0.0003109 ≤ Y ≤ 0.707
```



```
      ⍝  FIGURE 4
```

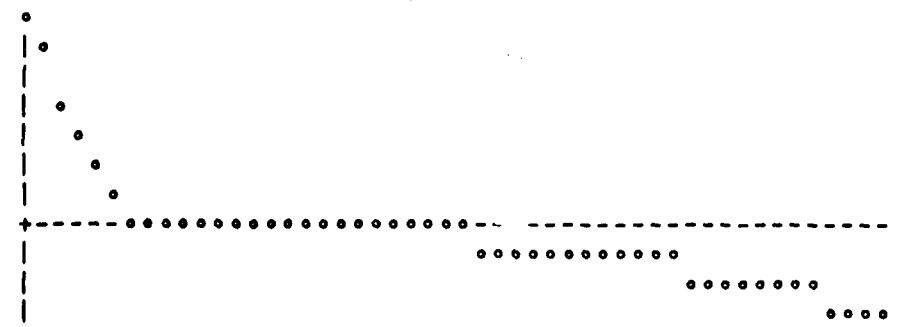
```
      ]CR'DMS1'
Z←N DMS1 S
⍝  TRUNCATE AT NTH NBRS
Z←1-((○S)*0.5)+1+2×+/[1]*-((⍳N)×⍳N)∘.+S←2×Ś×S
```

```
      10 PLOT 1 DMS1 1.5×S←(⍳50)+50
RANGE:  0.4453 ≤ Y ≤ 0.9248
```



```
      10 PLOT 2 DMS1 2×S
RANGE:  0.2602 ≤ Y ≤ 0.8997
```

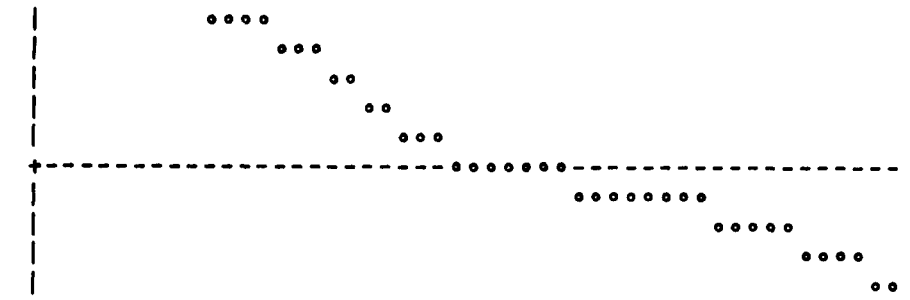
```
      10 PLOT 3 DMS1 3×S
RANGE:  0.3178 ≤ Y ≤ 0.8496
```



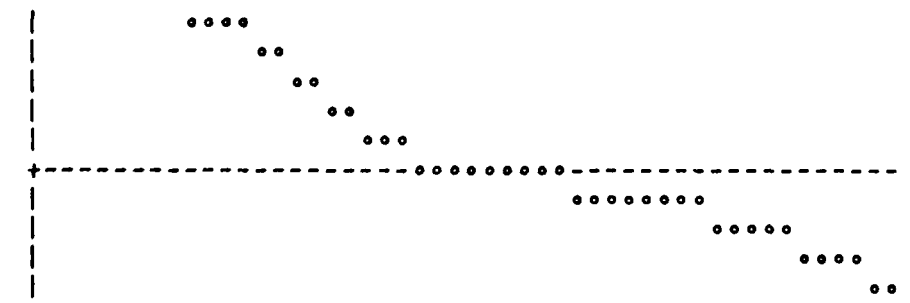
```
      ⍝  FIGURE 5
```

```
      ⎕CR'DHS2'
Z←N DHS2 S;NBR
⍝  HEX - NTH NBRS UP TO N=10 (12 AT R=21*÷2)
NBR← 1 3 4 7 9 12 13 16 19 21
NBR← 2 10 ⍴NBR, 6 6 6 12 6 6 12 6 12 12
Z←(0.5×3*0.5)-(○S)÷1+(N↑NBR[2;])+.×*-(N↑NBR[1;])∘.÷S÷2×S×S


      10 PLOT 1 DHS2 S←(⍳50)÷50
RANGE: ¯0.4883 ≤ Y ≤ 0.8635
ooooo
|    oooo
|        oo
|          oo
|            oo
|              ooo
+-----------------ooooooooo-----------------------
|                          oooooooooo
|                                    oooooo
|                                          oooo
|                                              ooo

      10 PLOT 2 DHS2 1.2×S
RANGE: ¯0.3638 ≤ Y ≤ 0.8624
oooo
|   ooo
|      oo
|        o
|         oo
|           oo
|             oo
+---------------oooooooooooooooo------------------
|                               ooooooooo
|                                        oooooo
|                                              ooo

      10 PLOT 3 DHS2 1.5×S
RANGE: ¯0.3794 ≤ Y ≤ 0.8604
ooo
|  oo
|    oo
|      o
|       oo
|         o
|          oo
+------------ooooooooooooooooooooo----------------
|                                 ooooooo
|                                        oooooo
|                                              ooo


      ⍝  FIGURE 6
```

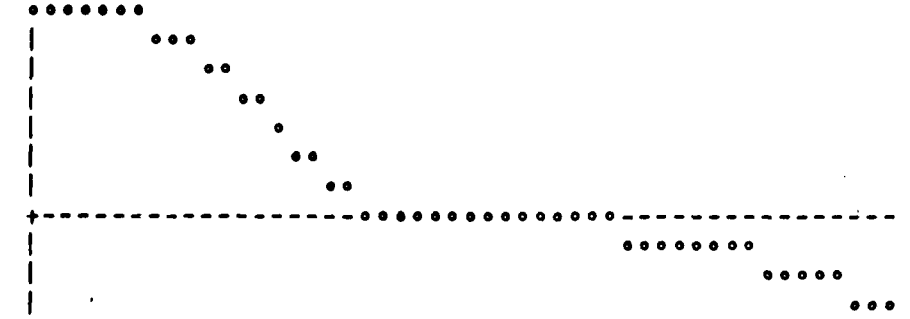
```
      ]CR'DI33'
Z←N DI33 S;NBR
⍝  PCC - NTH NBRS UP TO N=24 (R=5 - 3 WAYS)
NBR← 12 6 24 12 24 8 48 6 36 24 24 24
NBR←NBR, 72 0 48 12 48 30 72 24 48 24 48 8 84
N←N+N>13
Z←(*2*0.5)-((○S)*1.5)*1+(N↑NBR)+.××-(⍳N)∘.*S←2×S×S

      10 PLOT 1 DI33 0.75×S←(⍳50)÷50
RANGE:  0.4127 ≤ Y ≤ 0.7071
```



```
      10 PLOT 2 DI33 0.85×S
RANGE:  0.4295 ≤ Y ≤ 0.707
```

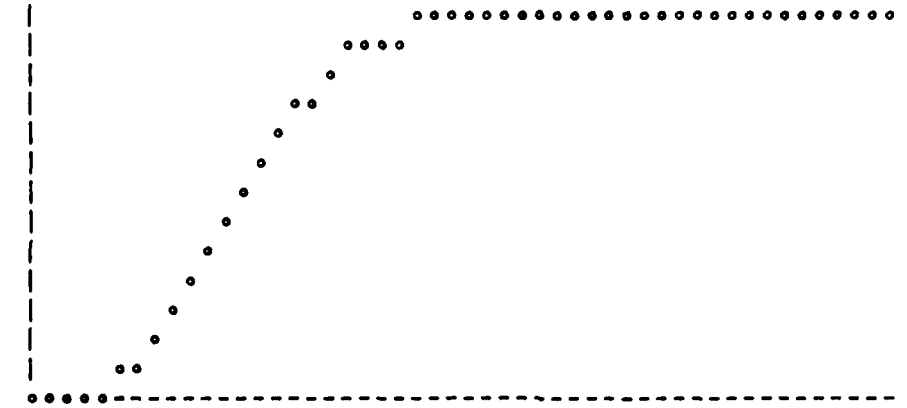


```
      10 PLOT 3 DI33 S
RANGE:  0.2871 ≤ Y ≤ 0.707
```

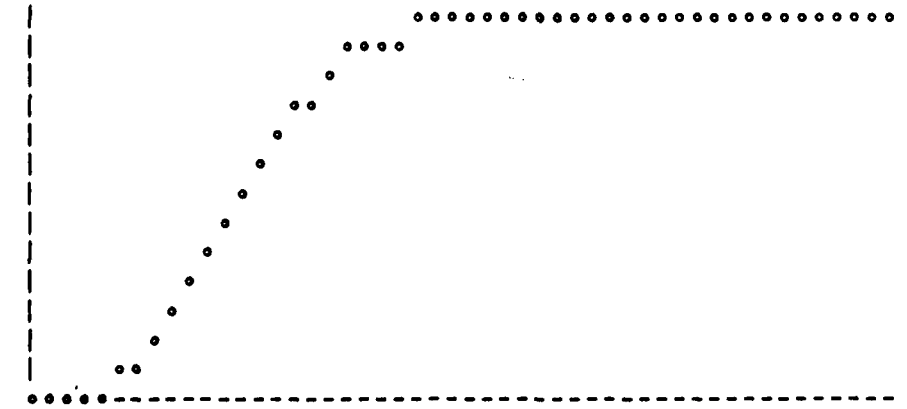


⍝ FIGURE 7

```
      ]CR 'CLERR'
Z←N CLERR S;S0;SS;RN;DR;K;S1;S2;S3;S4
⍝  TRUNCATE AT N1ST (PARTICLE) AND N2ND (CELL) NBRS
S0←( 0.57234 0.7014 0.8092 0.9037 )[N[1]]
RN←(1+2×+/[1]⍳-(N×N←⍳N[2])∘.+SS)+(0SS←(S0×S0)+S×S)*0.5
DR←(⍴S)⍴0,0⍴K←1+⍴N
BR1:→(∨/Z>DR×1E¯8)/BR1,0⍴K←K+1,0⍴DR←DR+2×Z←⍟-(K×K)÷SS
SS←(4×SS)-2×S0×S0,0⍴S2←(⍴S)⍴1,0⍴K←2,0⍴DR←DR+(0SS)*0.5
BR2:→(∨/Z>S2×1E¯8)/BR2,0⍴K←K+2,0⍴S2←S2+2×Z←⍟-(K×K)÷SS
S4←(⍴S)⍴0,0⍴K←1
BR3:→(∨/Z>S4×1E¯8)/BR3,0⍴K←K+2,0⍴S4←S4+2×Z←⍟-(K×K)÷SS
S1←1,0⍴K←2
BR4:→(Z>S1×1E¯8)/BR4,0⍴K←K+2,0⍴S1←S1+2×Z←⍟-K×K÷2×S0×S0
S3←0,0⍴K←1
BR5:→(Z>S3×1E¯8)/BR5,0⍴K←K+2,0⍴S3←S3+2×Z←⍟-K×K÷2×S0×S0
Z←(((S1×S2)+S3×S4)+0S0×(SS÷2)*0.5)-RN×RN+2×DR

      13 PLOT 1 5 CLERR S←(⍳50)÷50
RANGE: 1.925E¯7 ≤ Y ≤ 0.003111
```

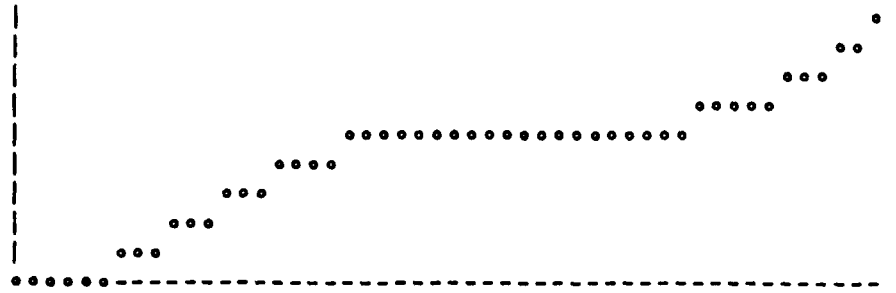


```
      13 PLOT 2 5 CLERR S
RANGE: 7.499E¯9 ≤ Y ≤ 0.0001212
```

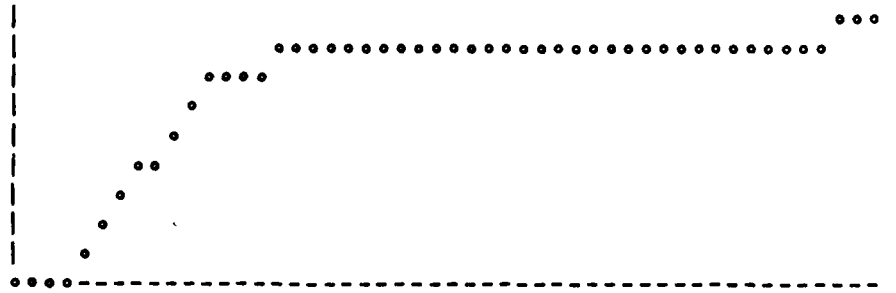


```
      ⍝  FIGURE 8
```

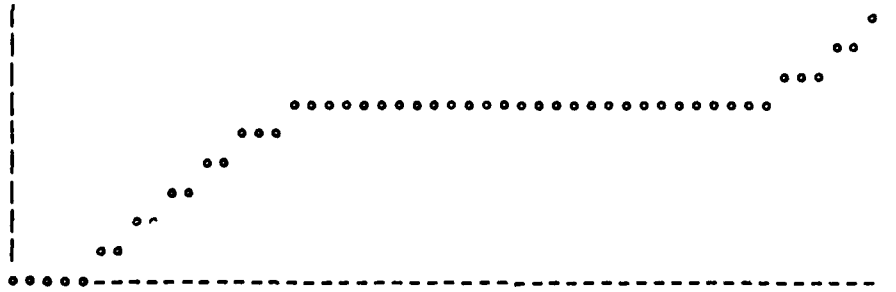
```
      9 PLOT 1 1 CLERR S←(⍳50)÷50
RANGE: 1.926E¯7 ≤ Y ≤ 0.005536
```



```
      9 PLOT 1 2 CLERR 1.5×S
RANGE: 9.65E¯7 ≤ Y ≤ 0.003633
```



```
      9 PLOT 2 2 CLERR 1.2×S
RANGE: 1.55E¯8 ≤ Y ≤ 0.0001837
```



```
      9 PLOT 2 3 CLERR 1.6×S
RANGE: 4.854E¯8 ≤ Y ≤ 0.0001342
```

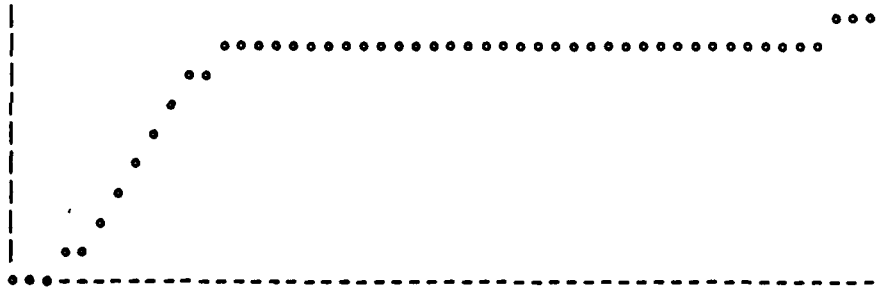


FIGURE 9

```
      ⎕CR'DRHO'
Z←N DRHO S;S0
⍝  TRUNCATE AT N1ST (PARTICLE) AND N2ND (CELL) NBRS
S0←S0×S0←( 0.57234 0.7014 0.8092 0.9037 )[N[1]]
Z←1-(1+2×+/[1]*-(N×N←⍳N[2])∘.+S)+(○S←S0+S×S)*0.5

      8 PLOT 1 1 DRHO S←(⍳50)+50
RANGE:  0.07856 ≤ Y ≤ 0.04924
|                                                    ∘∘∘
|                                                ∘∘∘∘∘
|                                          ∘∘∘∘∘∘∘
+-----------------------------∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘
|                    ∘∘∘∘∘∘∘
|                ∘∘∘∘
|            ∘∘∘∘
|        ∘∘∘∘
|    ∘∘∘∘
∘∘∘∘∘
      8 PLOT 1 2 DRHO 1.5×S
RANGE:  0.07818 ≤ Y ≤ 0.02286
|                                                    ∘∘∘
|                                              ∘∘∘∘∘∘∘∘∘
+--------------∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘
|              ∘∘∘
|           ∘∘∘
|        ∘∘
|     ∘∘
|  ∘∘
∘∘∘
      8 PLOT 2 2 DRHO 1.2×S
RANGE:  0.01548 ≤ Y ≤ 0.007904
|                                                      ∘∘
|                                                 ∘∘∘∘
|                                          ∘∘∘∘∘∘∘∘∘
+-----------------∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘
|               ∘∘∘∘∘
|            ∘∘∘
|        ∘∘∘
|     ∘∘∘
∘∘∘∘
      8 PLOT 2 3 DRHO 1.6×S
RANGE:  0.01541 ≤ Y ≤ 0.003599
|                                                    ∘∘∘
|                                       ∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘
+-----------------------------------∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘
|            ∘∘∘
|         ∘∘
|        ∘
|     ∘∘
|  ∘∘∘
∘∘

      ⍝ FIGURE 10
```

Difference Analogs of Hamiltonian Systems

Frederic Bisshopp

Division of Applied Mathematics
Brown University
Providence, Rhode Island 02912

Numerical integration of Hamiltonian systems brings forth the question of whether or not the difference analogs have an energy invariant that corresponds to the energy integral of the differential equations. Several difference analogs of the harmonic oscillator will be examined in that light, and then generalizations will be given.

The differential equations,

$$\dot{x} = p, \ \dot{p} = -x, \tag{1}$$

have the energy integral,

$$E = \frac{1}{2}(p^2 + x^2). \tag{2}$$

Action-angle variables for the harmonic oscillator are $J=2\pi E$ and θ, with

$$x = \sqrt{2E}\cos(\theta-\theta_o), \ p = -\sqrt{2E}\sin(\theta-\theta_o) \tag{3}$$

and

$$\dot{\theta} = 1, \ \dot{J} = 0. \tag{4}$$

Even the simplest difference analog,

$$\theta_k = kh, \ J_k = J_o, \tag{5}$$

gives exact values of the solution. The problem of interest

here has to do with difference analogs of the primitive equations (1).

Explicit difference analog:

$$x_{k+1} = x_k + hp_k, \quad p_{k+1} = p_k - hx_k \tag{6}$$

defines the iteration,

$$\begin{pmatrix} x \\ p \end{pmatrix} \leftarrow \begin{pmatrix} 1 & h \\ -h & 1 \end{pmatrix} \begin{pmatrix} x \\ p \end{pmatrix} \tag{7}$$

where the reversed arrow indicates new values that are assigned when $t \leftarrow t+h$. The eigenvalues are $1 \pm ih$ and their magnitude is $\sqrt{1+h^2}$. Thus the explicit analog is unstable; its phase-portraits spiral outward.

Implicit difference analog:

$$x_{k+1} - hp_{k+1} = x_k, \quad p_{k+1} + hx_{k+1} = p_k \tag{8}$$

defines the iteration,

$$\begin{pmatrix} x \\ p \end{pmatrix} \leftarrow \frac{1}{1+h^2} \begin{pmatrix} 1 & h \\ -h & 1 \end{pmatrix} \begin{pmatrix} x \\ p \end{pmatrix} . \tag{9}$$

The eigenvalues are $(1 \pm ih)/(1+h^2)$ and their magnitude is $1/\sqrt{1+h^2}$. The implicit analog is stable, but unsatisfactory since its phase-portraits spiral inward.

Central difference analog:

$$\begin{aligned} x_{k+1} - \tfrac{1}{2} hp_{k+1} &= x_k + \tfrac{1}{2} hp_k \\ p_{k+1} + \tfrac{1}{2} hx_{k+1} &= p_k - \tfrac{1}{2} hx_k \end{aligned} \tag{10}$$

defines the iteration,

$$\begin{pmatrix} x \\ p \end{pmatrix} \leftarrow \frac{1}{1 + \frac{1}{4} h^2} \begin{pmatrix} 1 - \frac{1}{4} h^2 & h \\ -h & 1 - \frac{1}{4} h^2 \end{pmatrix} \begin{pmatrix} x \\ p \end{pmatrix} \tag{11}$$

The matrix that transforms $(x,p)^T$ is a rotation, so $E = \frac{1}{2}(p_k^2+x_k^2)$ is an energy invariant of the central difference analog. The only source of truncation error is in the phase of the solution,

$$x_k = \sqrt{2E} \cos(k\tau-\theta_o), \quad p_k = -\sqrt{2E} \sin(k\tau-\theta_o), \tag{12}$$

where

$$\tan \tau = h/(1 - \tfrac{1}{4} h^2). \tag{13}$$

For the exact solution, τ would be h, as in equations (3,5); from equation (13)

$$\tau = h - \tfrac{1}{12} h^3 + 0(h^5) , \qquad |h| < 2. \tag{14}$$

The attractiveness of the central difference analog is diminished because it is an implicit scheme.

Leap-frog method:

$$x_{k+1} = x_{k-1}+2hp_k, \quad p_{k+1} = p_{k-1}-2hx_k \tag{15}$$

is a three-level, explicit, central difference scheme. It has solutions of the form

$$x_k = x_o\lambda^k, \quad p_k = p_o\lambda^k \tag{16}$$

where

$$\begin{vmatrix} \lambda^2-1 & 2h\lambda \\ -2h\lambda & \lambda^2-1 \end{vmatrix} = 0 \ . \qquad (17)$$

The eigenvalues are solutions of $\lambda^2 = 1-2h^2 \pm 2ih\sqrt{1-h^2}$ and their magnitude is 1 for $|h| \leq 1$. There is a well-behaved mode,

$$x_k = \sqrt{2E}\cos(k\tau-\theta_o),\ p_k = -\sqrt{2E}\,\frac{h}{\sin\tau}\sin(k\tau-\theta_o), \qquad (18)$$

and an ill-behaved mode,

$$x_k = A(-1)^k\cos(k\tau-\theta_o),\ p_k = A\,\frac{h}{\sin\tau}\,(-1)^k\sin(k\tau-\theta_o), \qquad (19)$$

where

$$\tan 2\tau = 2h\sqrt{1-h^2}/(1-2h^2). \qquad (20)$$

The phase error follows from

$$\tau = h + \frac{1}{6}h^3 + 0(h^5), \quad |h| < 1/\sqrt{2}\ , \qquad (21)$$

and, if the ill-behaved mode is suppressed, the energy invariant of the leap-frog method is a close approximation of the energy integral since $h/\sin\tau = 1+0(h^4)$.

Flip-flop integration:

The explicit, two-level scheme,

$$x_{k+1} = x_k+hp_k,\quad p_{k+1}+hx_{k+1} = p_k \qquad (22)$$

defines the iteration,

$$\begin{pmatrix} x \\ p \end{pmatrix} \leftarrow \begin{pmatrix} 1 & h \\ -h & 1-h^2 \end{pmatrix}\begin{pmatrix} x \\ p \end{pmatrix} \ . \qquad (23)$$

The eigenvalues are $(2-h^2 \pm ih\sqrt{4-h^2})/2$, and their magnitude is 1 for $|h| \leq 2$. The matrix that transforms $(x,p)^T$ clearly is not a rotation, but an energy invariant can be found by multiplying the second of equations (22) by $(p_{k+1}+p_k)/2$. The result,

$$E = \frac{1}{2}(p_k^2+hp_kx_k+x_k^2)$$

$$= \frac{1}{4}((1 + \frac{h}{2})(p_k+x_k)^2 + (1 - \frac{h}{2})(p_k-x_k)^2) \tag{24}$$

indicates that the solution lies on an ellipse at the angle $\theta_o-k\tau$, where

$$\tan \tau = h\sqrt{4-h^2}/(2-h^2) \tag{25}$$

The phase error follows from

$$\tau = h + \frac{1}{24} h^3 + 0(h^5), \quad |h| < \sqrt{2} \tag{26}$$

By comparison, the flip-flop integration is best at representing the phase of the oscillation, worst at representing the energy integral. Nevertheless, it has an energy invariant, and for the action-angle variables the relative errors are $h^2/4$ and $h^2/24$, respectively, when $(x_o,p_o) = (x_o,0)$ or $(0,p_o)$. By the more familiar reckoning, the relative error in the waveform is of $0(h)$.

The principal advantages of the flip-flop integration are that it is two-level and explicit, and it can easily be generalized for some Hamiltonian systems. Consider first

$$\dot{x} = p, \quad \dot{p} = f(x) \tag{27}$$

and the flip-flop integration

$$x_{k+1} = x_k + hp_k, \quad p_{k+1} = p_k + hf(x_{k+1}). \tag{28}$$

With one degree of freedom, $f(x)$ is derivable from the potential

$$V(x) = -\int^{x} f(z)dz, \tag{29}$$

and, by the trapezoid rule,

$$V(x_{k+1})-V(x_k) = -\frac{1}{2}(f(x_{k+1})+f(x_k))(x_{k+1}-x_k) + \frac{1}{12} f''(\xi)(x_{k+1}-x_k)^3, \tag{30}$$

where ξ lies in $[x_k, x_{k+1}]$. We define

$$E_k = \frac{1}{2} p_k^2 - \frac{1}{2} hp_k f(x_k) + V(x_k), \tag{31}$$

and it follows from equations (28,30) that

$$E_{k+1}-E_k = \frac{1}{12} h^3 p_k^3 f''(\xi). \tag{32}$$

It cannot be seen by this argument whether or not the flip-flop integration has an energy invariant, but there is a <u>near invariant</u> that confines phase portraits of solutions of the difference equations to a neighborhood of the energy integral, with relative errors of $0(h)$ for times as long as $0(1/h)$. A possible drift of $0(h^2)$ for times of $0(1)$ is more or less consistent with $0(h^2)$ errors in action-angle variables, and again there are relative errors of $0(h)$ in the waveform of an oscillation. It can be expected that equation (32) overestimates

the drift per cycle of a nonlinear oscillation.

Now let us consider Hamiltonian systems defined by $H(x,p) = T(p)+V(x)$, where x and p are vectors. Let

$$g_k \equiv \nabla T|_{p=p_k} , \quad f_k \equiv -\nabla V|_{x=x_k} . \tag{33}$$

From the Taylor series of T and V, expanded about (x_k,p_k) and (x_{k+1},p_{k+1}), it follows that

$$\begin{aligned} H(x_{k+1},p_{k+1})-H(x_k,p_k) &= \tfrac{1}{2}(g_k+g_{k+1})^T(p_{k+1}-p_k) \\ &\quad - \tfrac{1}{2}(f_k+f_{k+1})^T(x_{k+1}-x_k)+0(|x_{k+1}-x_k|^3,|p_{k+1}-p_k|^3) \end{aligned} \tag{34}$$

where $|\xi|^2 = \xi^T\xi$. It follows easily that there is a near invariant (to $0(h^3)$) for either of the flip-flop integrations,

$$\begin{aligned} &x_{k+1} = x_k+hg_k , \quad p_{k+1} = p_k+hf_{k+1}, \\ &E_k \equiv T(p_k)+V(x_k) - \tfrac{1}{2}\, hg_kf_k, \end{aligned} \tag{35}$$

<u>or</u>

$$\begin{aligned} &p_{k+1} = p_k+hf_k, \quad x_{k+1} = x_k+hg_{k+1} , \\ &E_k \equiv T(p_k)+V(x_k) + \tfrac{1}{2}\, hg_kf_k. \end{aligned} \tag{36}$$

The argument fails for the general $H(x,p)$.

As an example, the cubic nonlinear oscillator,

$$\ddot{x} + x^3 = 0, \tag{37}$$

has been simulated by flip-flop integration,

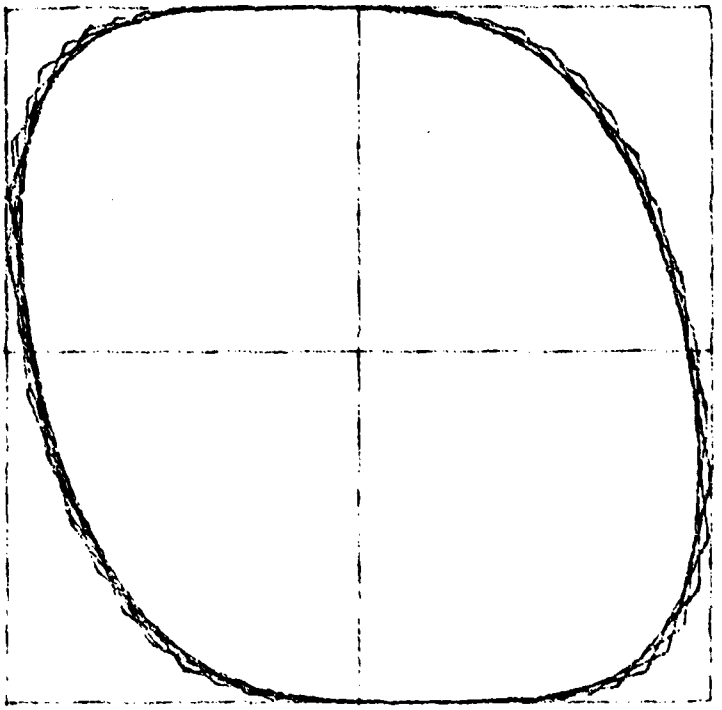
```
[1]   x ← x + hp
[2]   p ← p - hx*3                                    (38)
[3]   → 1
```

The results, as illustrated in the accompanying figure, suggest that although the near invariant is not strictly conserved there may be no systematic drift after all. The phase portrait of approximately twenty cycles of the oscillation appears to be confined to a band (probable width of $0(h^2)$) centered about the near invariant. The value of h in the computation is 0.5 —— for most, but not all values of h greater than a number near 0.75, the algorithm is unstable.

Figure caption:

Phase portrait, $\dot{x}$ vs. x for $0 \leq t \leq 125$ in steps of 0.5.

Initial data: $x(0) = 1$, $\dot{x}(0) = 0$.



Appendix C: Flow in a channel by biased differences

The flow that is simulated here occupies a rectangle with nodal coordinates

Y = MB, MM, ..., 2, 1 (bottom to top)

X = 1, 2, ..., NN, NB (left to right)

The sets of indices that are used to define central nodes and their neighbors are

N1 = 1, ..., NN-1

NO = 2, ..., NN

N3 = 3, ..., NB

M_2 = 1, ..., MM-1

MO = 2, ..., MM

M4 = 3, ..., MB

Thus the correspondence between index-pairs and nodes is

Central	⟷	(MO,NO)
Western	⟷	(MO,N1)
Northern	⟷	(M2,NO)
Eastern	⟷	(MO,N3)
Southern	⟷	(M4,NO)

The input variable, u, and output, z, are 5 by MB by NB arrays of values of u_1, u_2, f_1, f_2 and π at times t and t+k. The parameter Nu is actually $\nu k/h^2$ since length and time are measured in units of h and k to eliminate unnecessary multiplications.

The first part of the program carries out the forward integration of $\dot{X} = u$ as described in Section 3 of this report. The second and third parts are decoupled from one another in this early version of the method because only the hydrostatic component of the pressure boundary conditions has been included.

Thus the boundary conditions are:

Western boundary (inflow)

$\pi_c = \pi_e - hf_1(\underline{x}_c)$ (hydrostatic)

$\underline{u}_c = (\text{constant}, 0)$ (slug flow)

Eastern boundary (outflow)

$\pi_c = \text{constant}$

$\underline{u}_c = (u_{1w}, 0)$ (fully developed)

Northern boundary (rigid)

$\pi_c = \pi_s + hf_2(\underline{x}_c)$ (hydrostatic)

$\underline{u}_c = (0,0)$ (no slip)

Southern boundary (center line)

$\pi_c = \pi_n - hf_2(\underline{x}_c)$ (hydrostatic)

$\underline{u}_c = (u_{1n}, 0)$ (symmetric)

The last part of the program was added to correct for global truncation error. At each station along the x_1-axis u_1 is renormalized to preserve the constant total mass flux.

The initial condition was slug flow everywhere but at the rigid boundary of an 11 by 31 node grid. The initial value of u_1 and the inflow velocity were set at $1/\sqrt{2}$ to insure that no bias lines would fall outside local skeletons, and the body force was set to zero. Nu ($= \nu k/h^2$) was set at 0.2, and after 15 time steps (CPU time, 12 sec.) the flow was almost at a steady state. Figure 1 shows the profiles of u_1 at the horizontal stations 1, 3, 5, 7 and 9. The numbers above and below each graph are the maximum and minimum values of u_1. In figure 2 the same profiles are shown after 10 more time steps. In figures 3 - 5 the value $u_2 = 0$ at the rigid boundary is not included and the

scale is expanded to show detailed structure of the velocity profiles. The progression of the point of maximum velocity from edge to center as one moves downstream is consistent with results of other numerical simulations and with results of boundary layer theory.

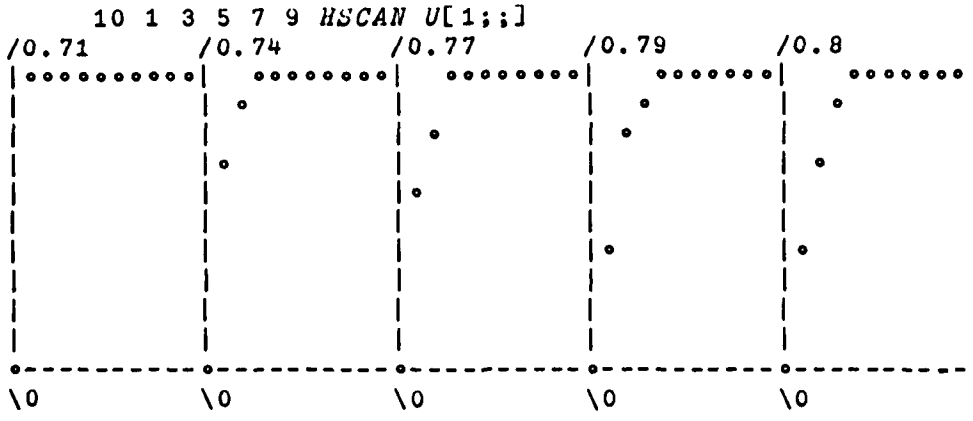


FIGURE 1: *VELOCITY PROFILES*
STATIONS 1,3,5,7,9 *AFTER* 15 *TIME STEPS*

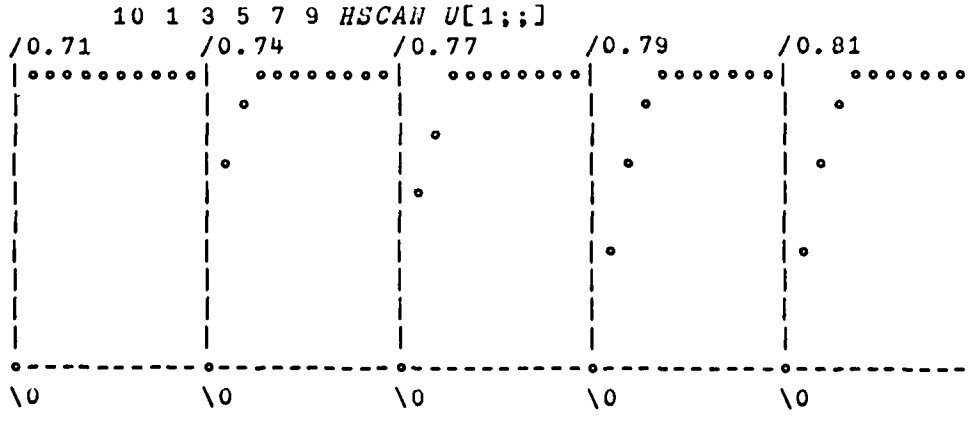


FIGURE 2; *VELOCITY PROFILES*
STATIONS 1,3,5,7,9 *AFTER* 25 *TIME STEPS*

FIGURE 3: STATIONS 1 TO 8 AFTER 25 TIME STEPS

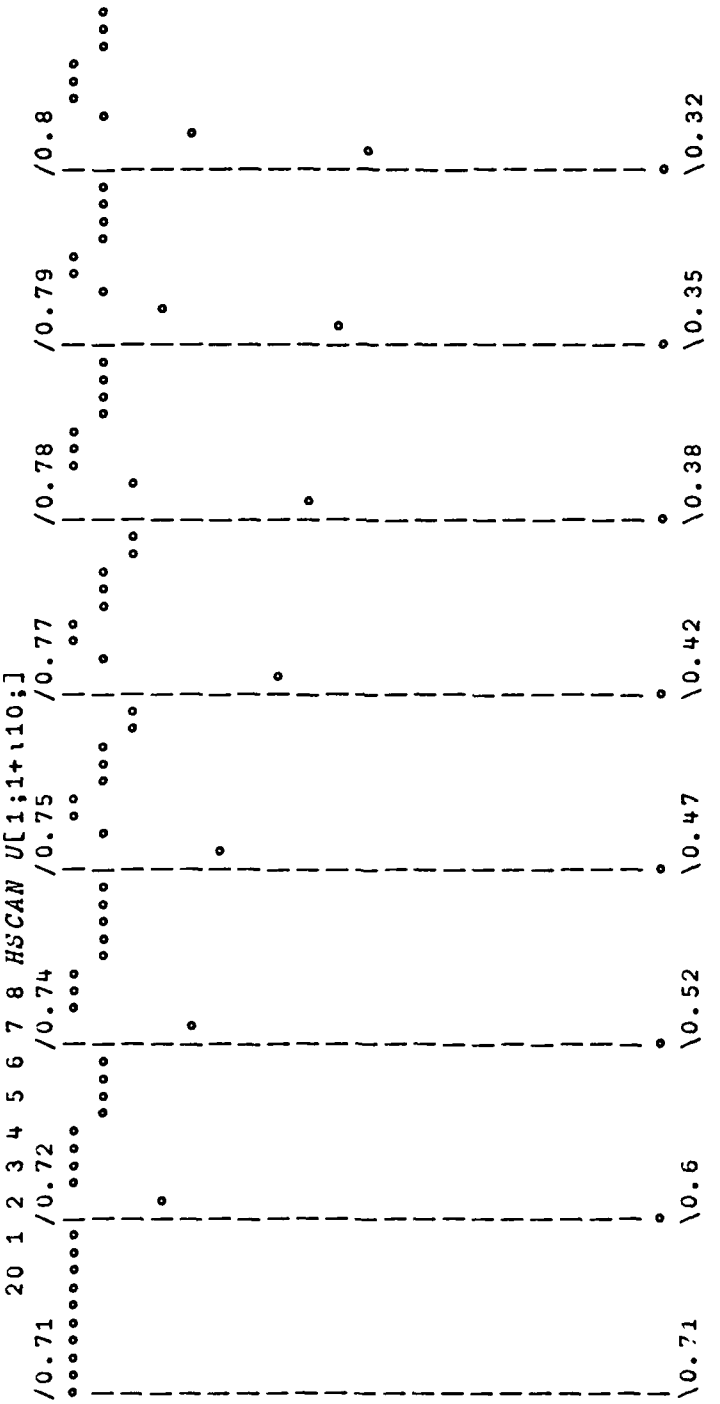
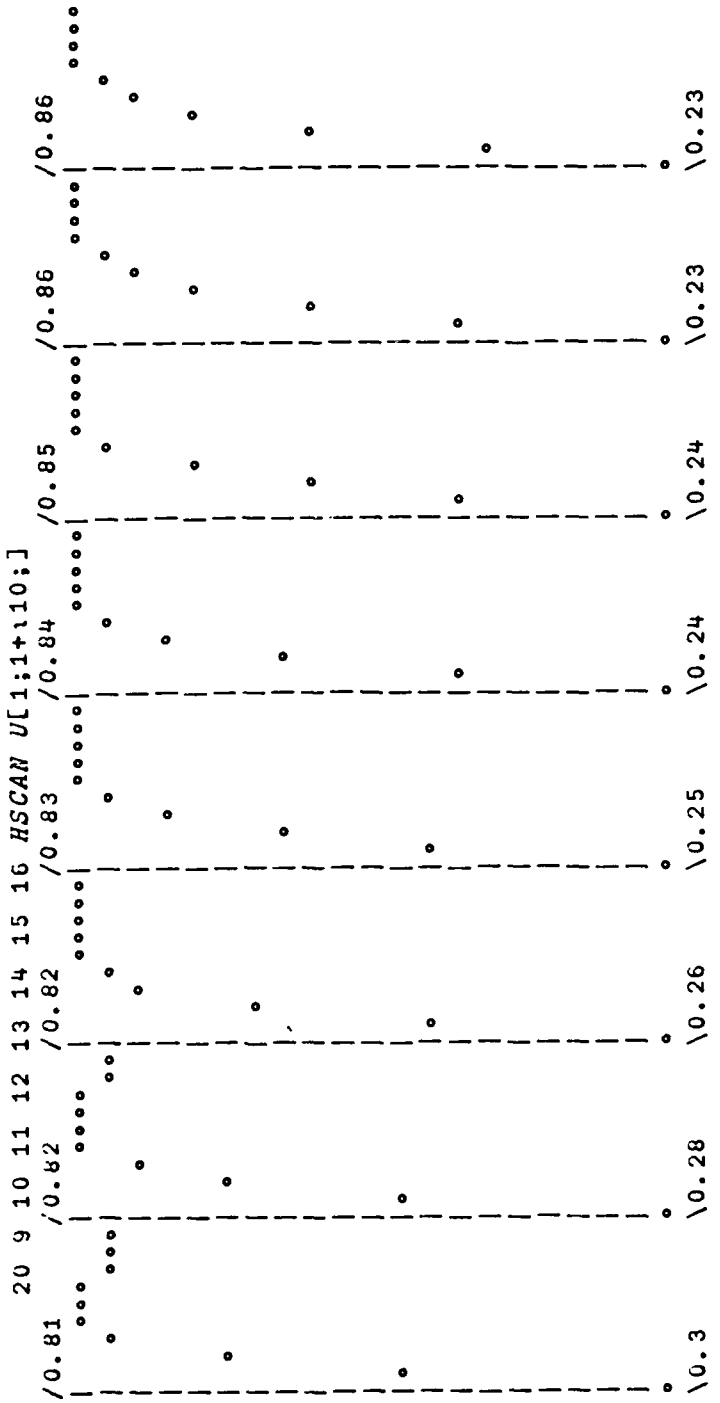


FIGURE 4: STATIONS 9 TO 16 AFTER 25 TIME STEPS



```
      ∇TWIF[□]∇
      ∇ Z←NU TWIF U;MB;MM;M;M2;M0;M4;NB;NN;N;N3;N0;N1;K
[1]   ⍝  TWO-DIMENSIONAL, INCOMPRESSIBLE FLOW
[2]   ⍝  U=U[1 2;;], F=U[3 4;;], P=U[5;;], ΔT=ΔX=1, |U|≤1
[3]   ⍝  SET UP INDICES
[4]    MB←1+MM←1+M←⍴M2←1+M0←1+M4←⍳¯2+(⍴U)[2]
[5]    NB←1+NN←1+N←⍴N3←1+N0←1+N1←⍳¯2+(⍴U)[3]
[6]   ⍝  FORWARD INTEGRATION, X'=U
[7]    K←U[1,2;M0;N0]
[8]    Z←(K-(U[1,2;M0;N1]×K[1,1;;]>0)+U[1,2;M0;N3]×K[1,1;;]<0)××K[1,1;;]
[9]    Z←Z,[1](K-(U[1,2;M2;N0]×K[2,2;;]>0)+U[1,2;M4;N0]×K[2,2;;]<0)××K[2,2;;]
[10]   Z←Z,[1]((Z[1;;]←1+Z[1;;])×Z[4;;]←1+Z[4;;])-Z[2;;]×Z[3;;]
[11]   U[1,2;M0;N0]←((Z[4,1;;]×K)-Z[3,2;;]×⌽K)÷Z[5,5;;]
[12]  ⍝  POISSON EQUATION FOR PRESSURE
[13]   K←SETK
[14]   Z←0.5×+/[1] U[1,3;M0;N3]+U[2,4;M4;N0]-U[1,3;M0;N1]+U[2,4;M2;N0]
[15]  BR1:U[5;M0;N0]←(U[5;M0;N1]+U[5;M2;N0]+U[5;M0;N3]+U[5;M4;N0]-Z)÷4
[16]   U[5;M0;1]←U[5;M0;2]-U[3;M0;1]
[17]   U[5;1,MB;1,2,N3]←U[5;2,MM;1,2,N3]+U[4;1;1,2,N3],[1]-U[4;MB;1,2,N3]
[18]   →(0<K←K-1)/BR1
[19]  ⍝  BACKWARD INTEGRATION, U'+∇P=F+NU×ΔU
[20]   K←SETK
[21]   Z←U[1,2;M0;N0]+U[3,4;M0;N0]-0.5×(U[5;M0;N3]-U[5;M0;N1]),[0.5] U[5;M4;N0]-U[5;M2;N0]
[22]  BR2:U[1,2;M0;N0]←(Z+NU×U[1,2;M0;N1]+U[1,2;M2;N0]+U[1,2;M0;N3]+U[1,2;M4;N0])÷1+4×NU
[23]   U[1,2;M0;NB]←U[1,2;M0;NN]
[24]   U[1;MB;1,2,N3]←U[1;MM;1,2,N3]
[25]   →(0<K←K-1)/BR2
[26]  ⍝  BRUTE FORCE
[27]   U[1;;]←U[1;;]×((MB,NB)⍴+/[1] U[1;;1])÷(MB,NB)⍴+/[1] U[1;;]
[28]   Z←U
       ∇
```

Appendix D: Thermal instability

The problem that was used to check the biased difference algorithm for thermal convection was the Bénard problem. Thermal instability of a horizontal layer of a fluid, heated from below, is governed by the equations

$$\dot{\underline{X}} = \underline{u}$$

$$\dot{\underline{u}} = -\underline{\nabla}\pi + \alpha g\hat{z}(T - \langle T\rangle) + \nu\Delta\underline{u}$$

$$\underline{\nabla}\cdot\underline{u} = 0$$

$$\dot{T} = \kappa\Delta T$$

where

$$\pi = \frac{p}{\rho_o} + gz \ ,$$

α is the coefficient of thermal expansion, g is the acceleration of gravity, ν is the kinematic viscosity, κ is the thermal diffusivity, and $\langle T\rangle$ is the mean temperature. The boundary conditions that were used are

$\underline{u} = 0$ all boundaries

$T = T_o$ and T_1 lower and upper boundaries

$\frac{\partial T}{\partial n} = 0$ sidewalls

Two- and three-dimensional cases were implemented. For simplicity we will discuss the two-dimensional case; the three-dimensional algorithm is identical in form. Let the subscripts 1, 2 denote the components of $\underline{x} = (y,z)$, and let the subscripts c, n, s, u, d denote central, northern, southern, upward, and downward nodes in a rectangular grid. In the APL function TW1C the 4 by M by N dimensional array U contains nodal values of $\underline{u}$, T and π. North is to the right and up is up in the arrays; the correspondence between index pairs and directions is

$U_c \longleftrightarrow U[;MO;NO]$

$U_s \longleftrightarrow U[;MO;N1]$ $\quad N1 = NO-1$

$U_u \longleftrightarrow U[;M2;NO]$ $\quad M2 = MO-1$

$U_n \longleftrightarrow U[;MO;N3]$ $\quad N3 = N1+2$

$U_d \longleftrightarrow U[;M4;NO]$ $\quad M4 = M2+2$

The first part of the algorithm takes input values $\underline{u}$, T, π at time t (say); the parameters P = ν, κ, αg are not used until later. The forward integration of $\dot{\underline{X}} = \underline{u}$ with time increment k gives

$$\underline{x}_x = \bar{\underline{x}}_c + k\underline{u}(\bar{\underline{x}}_c)$$

where $\bar{\underline{x}}_c$ is the position at time t of the bias line that passes through $\underline{x}_c$ at time t+k, and $\underline{u}$ still refers to time t. With linear interpolation for $\underline{u}(\bar{\underline{x}}_c)$ the displacement

$$\bar{\underline{x}}_c = \underline{x}_c + \underline{\delta}$$

is given by the linear equations

$$(I + kM)\underline{\delta} = - k\underline{u}_c$$

where the space increment is h and

$$M_{11} = \frac{1}{h}((u_{1n} - u_{1c})(u_{1c} > 0) + (u_{1c} - u_{1s})(u_{1c} < 0))$$

$$M_{12} = \frac{1}{h}((u_{1u} - u_{1c})(u_{2c} > 0) + (u_{1c} - u_{1d})(u_{2c} < 0))$$

$$M_{21} = \frac{1}{h}((u_{2n} - u_{2c})(u_{1c} > 0) + (u_{2c} - u_{2s})(u_{1c} < 0))$$

$$M_{22} = \frac{1}{h}((u_{2u} - u_{2c})(u_{2c} > 0) + (u_{2c} - u_{2d})(u_{2c} < 0)) \ .$$

Note, the parenthetic expressions, $(u_{ic} < 0)$ and $(u_{ic} > 0)$ which have logical values 0 or 1, are used to compute arrays of coefficients of M in the APL function.

Given arrays of $\underline{\delta}$ at the central nodes, the initial data for velocity and temperature is

$$\overline{\underline{u}}_c = \underline{u}_c + M\underline{\delta}$$

$$\overline{T}_c = T_c + \underline{m}\cdot\underline{\delta}$$

where the vector $\underline{m}$ has components

$$m_1 = \frac{1}{h}((T_n - T_c)(u_{1c} > 0) + (T_c - T_s)(u_{1c} < 0))$$

$$m_2 = \frac{1}{h}((T_u - T_c)(u_{2c} > 0) + (T_c - T_d)(u_{2c} < 0)).$$

At this point the approximation

$$(\underline{\nabla}\cdot\overline{\underline{u}})_c = \frac{1}{2h}(\overline{u}_{1n} - \overline{u}_{1s} + \overline{u}_{2u} - \overline{u}_{2d})$$

is evaluated for use in the pressure equation.

The backward integration of the temperature equation defines the iteration that initiates the second part of the algorithm

$$T_c = (\overline{T}_c + \frac{\kappa k}{h^2}(T_n + T_s + T_u + T_d))/(1 + \frac{4\kappa k}{h^2}) \ .$$

Substitution of $\underline{\nabla}\cdot\underline{u} = 0$ in the divergence of the backward time integration of the momentum equation gives the pressure equation -- also iterated

$$\pi_c = \tfrac{1}{4}(\pi_n + \pi_s + \pi_u + \pi_d + \frac{\alpha g h}{2}(T_u - T_d) - \frac{h^2}{k}(\underline{\nabla}\cdot\overline{\underline{u}})_c) \ .$$

The momentum equations,

$$u_{1c} = [\overline{u}_{1c} + \frac{k}{2h}(\pi_s - \pi_n) + \frac{\nu k}{h^2}(u_{1n} + u_{1s} + u_{1u} + u_{1d})]/(1 + \frac{4\nu k}{h^2})$$

$$u_{2c} = [\overline{u}_{2c} + \frac{k}{2h}(\pi_d - \pi_u) + gk(T_c - \langle T\rangle) + \frac{\nu k}{h^2}(u_{2n} + u_{2s} + u_{2u} + u_{2d})]/(1 + \frac{4\nu k}{h^2})$$

complete the algorithm at internal nodes.

The temperature is fixed (unchanged) at the top and bottom boundaries, $T_c = T_n$ at the south boundary and $T_c = T_s$ at the north. Both components of $\underline{u}$ are zero on all boundaries, and the pressure boundary conditions are

Top:

$$\pi_c = \pi_d + \tfrac{1}{2}\alpha g h(T_c + T_d - 2\langle T\rangle) + \frac{2\nu}{h}u_{2d} + \frac{h}{4k}(\bar{u}_{2d} - u_{2d})$$

Bottom:

$$\pi_c = \pi_u - \tfrac{1}{2}\alpha g h(T_c + T_u - 2\langle T\rangle) - \frac{2\nu}{h}u_{2u} - \frac{h}{4k}(\bar{u}_{2u} - u_{2u})$$

North:

$$\pi_c = \pi_s + \frac{2\nu}{h}u_{1s} + \frac{h}{4k}(\bar{u}_{1s} - u_{1s})$$

South:

$$\pi_c = \pi_n - \frac{2\nu}{h}u_{1n} - \frac{h}{4k}(\bar{u}_{1n} - u_{1n}).$$

The small terms of $\frac{h}{4k}(\bar{\underline{u}} - \underline{u})$ were <u>not</u> included in the versions of the algorithm that were used here.

Finally, the last part of the algorithm renormalizes horizontal and vertical components of velocity to account for $\underline{\nabla}\cdot\underline{u} = 0$ globally, and sets the undetermined constant in the pressure field at a convenient value.

The onset of thermal instability in a fluid layer of depth d occurs at sufficiently high values of the Rayleigh number

$$R = \frac{\alpha g \Delta T d^3}{\kappa\nu}$$

where ΔT is temperature at bottom minus temperature at top. For an infinite layer bounded above and below by rigid boundaries the critical Rayleigh number is 1708, and for R greater than that convection appears. The fluid motion is in vortices

with alternating clockwise and counterclockwise circulation (rolls), and the width of a vortex is very slightly greater than its depth, d.

In the simulations we have run, time and distance are measured in units of k and h, i.e. $\Delta t = \Delta x = 1$. Thus an M by N array of nodes has a width (N-1) and a depth (M-1), and the Rayleigh number is

$$R = \frac{\alpha g \Delta T (M - 1)^3}{\kappa \nu}$$

For the two-dimensional cases the grid is 11 by 30, $\kappa = \nu = \Delta T = 1$, and αg is set at 2 or 4 to give Rayleigh numbers, 2000 or 4000. Runs were started with a uniform temperature gradient and small, randomly chosen velocity perturbation. Features of the resulting steady states that developed are shown in Figures 1 - 6.

In figures 1 - 3, R is 2000 and there are two weak primary cells and a very weak secondary cell. This is in accord with stability theory which predicts that neither two nor three cells can fill a container with aspect ratio 3:1 at slightly supercritical Rayleigh numbers. As far as we know, there is no theoretical work that has predicted that the resolution of that would be a secondary cell that is driven by the primary motion rather than by buoyancy.

In figures 4 - 6, R is 4000 and no secondary cell is needed to fill the container. This, incidentally, is about the limit of what can be simulated with $\Delta t = \Delta x$. When a component of uk/h exceeds one, the bias line falls outside the local skeleton, interpolations become extrapolations, and the method begins to be unreliable and possibly unstable.

Finally, a three-dimensional cell was simulated at Rayleigh number 4000 in an 11 by 11 by 11 node container. The alignment of the circulation with its axis on a diagonal in plan view allows a cell that has a width that is slightly greater than its height. Figure 7 is a plan view of the contours of w at $z = 6$ accompanied by elevation views of the direction fields of u and w at $y = 6$ (below) and v and w at $x = 6$ (to the left).

Altogether, biased differences appear to be a reliable and efficient way to simulate thermal convection. CPU time on the IBM 370/158 at Brown University is 5 seconds per time step of the three-dimensional simulation on 1331 nodes.

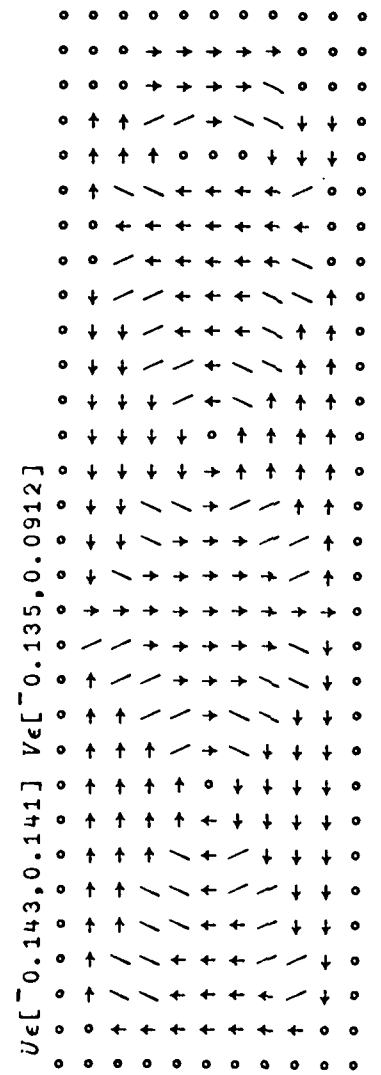


FIGURE 1: DIRECTION FIELD FOR R=2000

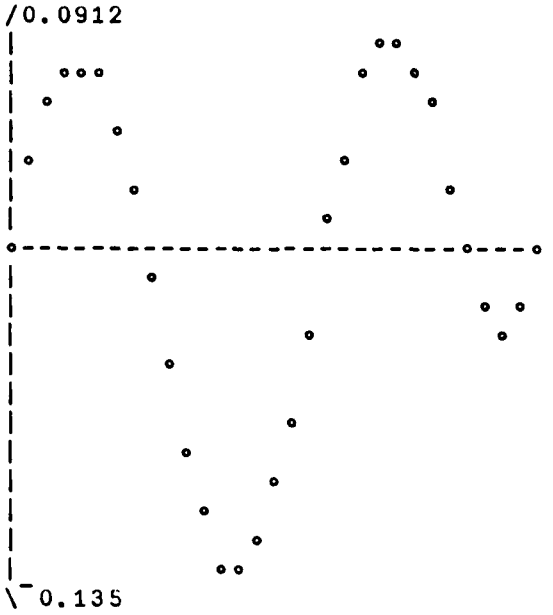


FIGURE 2: R=2000
VERTICAL VELOCITY AT MIDHEIGHT

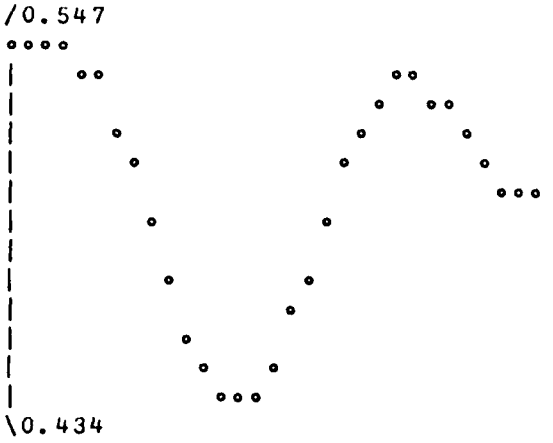


FIGURE 3: R=2000
TEMPERATURE AT MIDHEIGHT

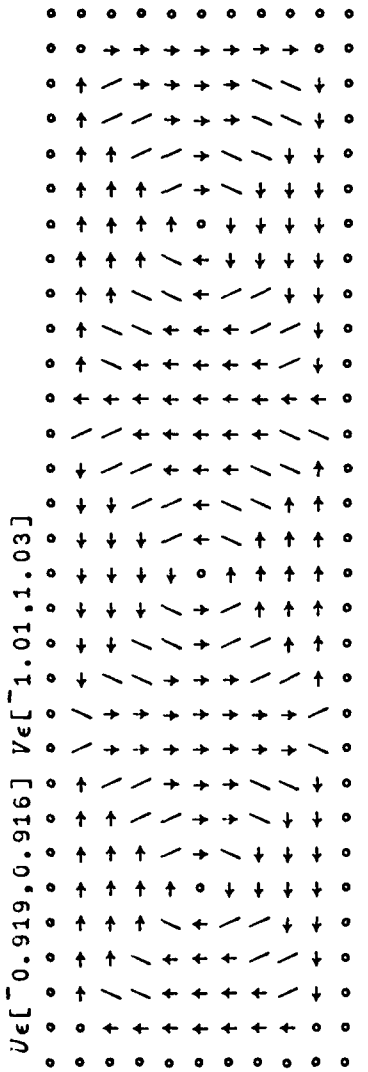


FIGURE 4: DIRECTION FIELD FOR R=4000

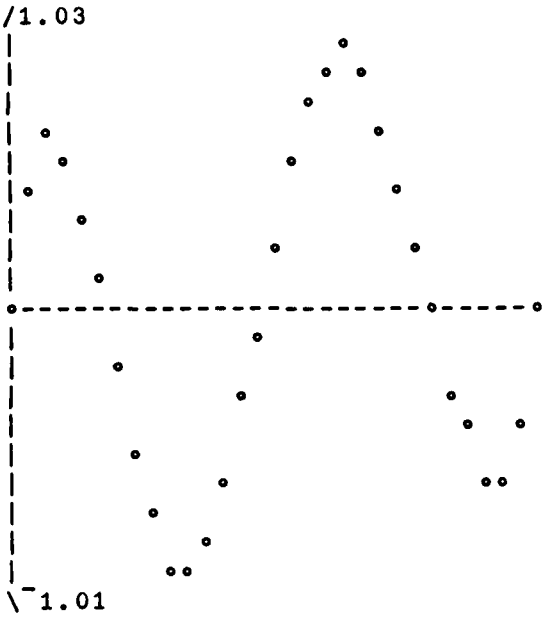


FIGURE 5: *R*=4000
VERTICAL VELOCITY AT MIDHEIGHT

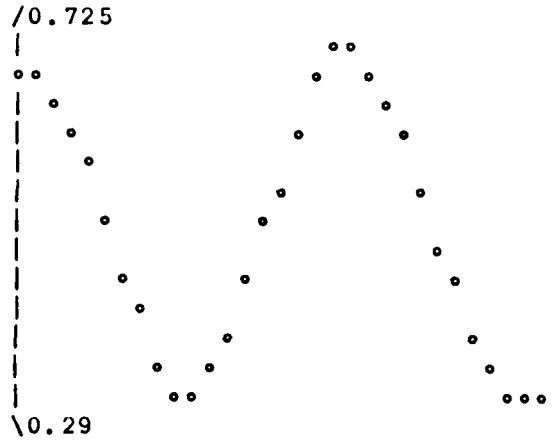


FIGURE 6: *R*=4000
TEMPERATURE AT MIDHEIGHT

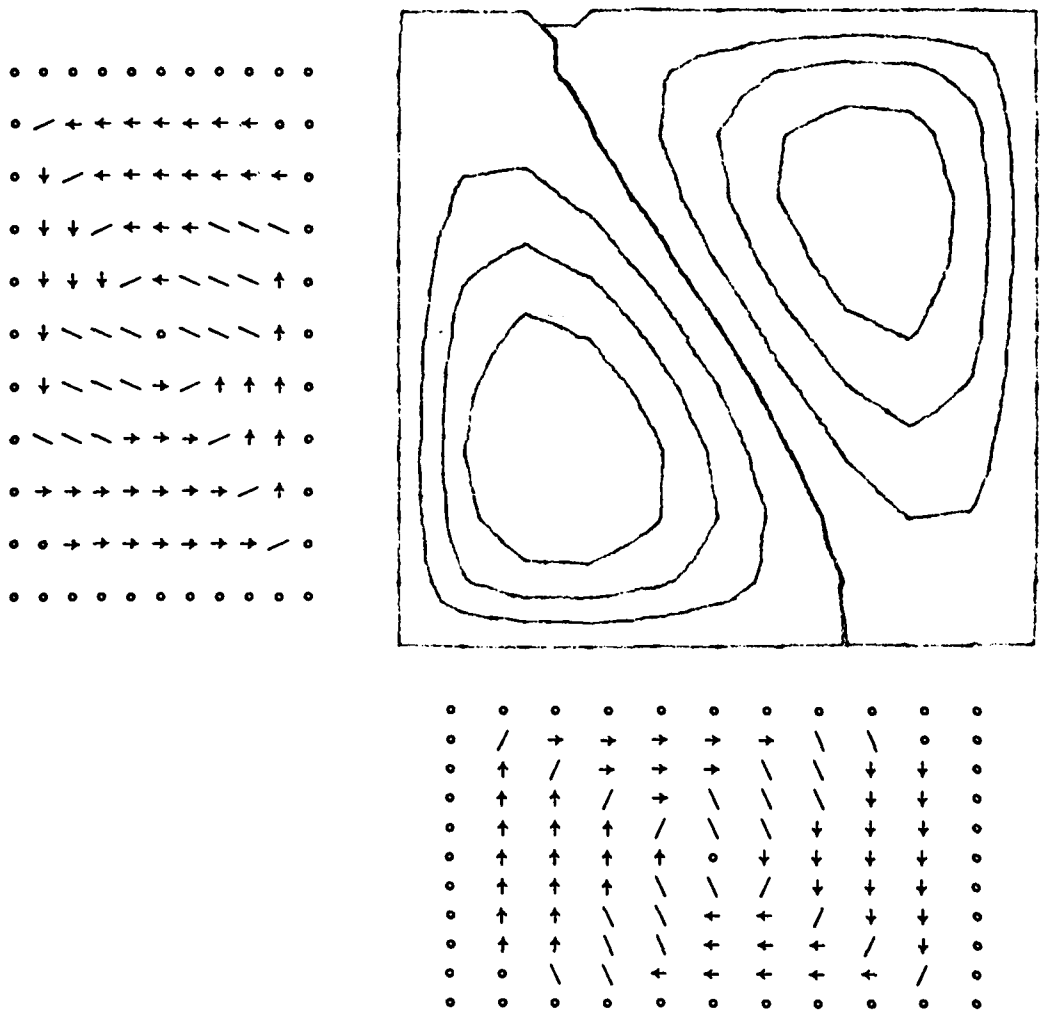


FIGURE 7: PLAN VIEW OF W AT MIDHEIGHT
ELEVATIONS OF DIRECTION FIELDS
AT MIDSTATIONS

```
      ⎕CR'TWIC'
Z←P TWIC U;MB;MM;M;M2;M0;M4;NB;NN;N;N1;N0;N3;V;K;F
⍝  TWO-DIMENSIONAL, INCOMPRESSIBLE CONVECTION
⍝  ΔT=ΔX=1, U[⍳4;;]=U,T,P
⍝  P=NU,KAPPA,ALPHA×G
⍝  SET UP INDICES
MB←1+MM←1+M←⍴M2←1+M0←1+M4←⍳¯2+(⍴U)[2]
NB←1+NN←1+N←⍴N3←1+N0←1+N1←⍳¯2+(⍴U)[3]
⍝  BIASED FORWARD DIFFERENCE FOR V[⍳3;⍳M;⍳N],[1] DIV V AT T=0
V←U[1 2 ;M0;N0]
Z←(V-(U[1 2 ;M0;N1]×V[1 1 ;;]>0)+U[1 2 ;M0;N3]×V[1 1 ;;]<0)××V[1 1 ;;]
Z←Z,[1](V-(U[1 2 ;M2;N0]×V[2 2 ;;]>0)+U[1 2 ;M4;N0]×V[2 2 ;;]<0)××V[2 2 ;;]
V←((Z[4 1 ;;]×V)-Z[3 2 ;;]×⊖V)÷(2,M,N)⍴((Z[1;;]←1+Z[1;;])×Z[4;;]←1+Z[4;;])-Z[2;;]×Z[3;;]
V←V,[1] U[3;M0;N0]-V[1;;]×(U[3;M0;N3]×V[1;;]<0)-(U[3;M0;N1]×V[1;;]>0)-U[3;M0;N0]××V[1;;]
V[3;;]←V[3;;]-V[2;;]×(U[3;M4;N0]×V[2;;]<0)-(U[3;M2;N0]×V[2;;]>0)-U[3;M0;N0]××V[2;;]
V←V,[1] 0.5×V[1;;1+N1,N]+V[2;1,¯1+M4;]-V[1;;1,¯1+N1]+V[2;1+M4,M;]
V[4;;1,N]←V[4;;1,N]+(2×V[1;;1],-V[1;;N])-0.5×V[1;;2,N]-V[1;;1,¯1+N]
V[4;1,M;]←V[4;1,M;]-(2×V[2;1;],[1]-V[2;M;])+0.5×V[2;1,¯1+M;]-V[2;2,M;]
⍝  BIASED BACKWARD DIFFERENCE FOR U[⍳4;;] AT T=1
K←2
KBR:U[3;M0;N0]←(V[3;;]+P[2]×U[3;M0;N1]+U[3;M2;N0]+U[3;M0;N3]+U[3;M4;N0])÷1+4×P[2]
U[3;M0;1,NB]←U[3;M0;2,NN]
F←P[3]×U[3;;]-(+/+/U[3;;])÷MB×NB
U[4;M0;N0]←(U[4;M0;N1]+U[4;M2;N0]+U[4;M0;N3]+U[4;M4;N0]-V[4;;]+0.5×F[M4;N0]-F[M2;N0])÷4
U[4;1,MB;N0]←U[4;2,MM;N0]+0.5×(+/[1] F[1 2 ;N0]),[0.5]-+/[1] F[MM,MB;N0]
U[4;1,MB;N0]←U[4;1,MB;N0]+(2×P[1])×U[2;2;N0],[1]-U[2;MM;N0]
U[4;M0;NB,1]←U[4;M0;NN,2]+(2×P[1])×U[1;M0;NN],-U[1;M0;2]
Z←V[1 2 ;;]+(0.5×U[4;M0;N1]-U[4;M0;N3]),[0.5] F[M0;N0]+0.5×U[4;M2;N0]-U[4;M4;N0]
U[1 2 ;M0;N0]←(Z+P[1]×U[1 2 ;M0;N1]+U[1 2 ;M2;N0]+U[1 2 ;M0;N3]+U[1 2 ;M4;N0])÷1+4×P[1]
→(0<K←K-1)/KBR
⍝  BRUTE FORCE
U[1 2 ;M0;N0]←U[1 2 ;M0;N0]-((M,N)⍴(+/[1] U[1;M0;N0])÷M),[0.5]⍉(N,M)⍴(+/U[2;M0;N0])÷N
U[4;1,MB;1,NB]←0.5×U[4;2,MM;1,NB]+U[4;1,MB;2,NN]
U[4;;]←U[4;;]-0.5×(⌈/⌈/U[4;;])+⌊/⌊/U[4;;]
Z←U
```

Appendix F: Interactive generation of contour maps

The APL function CONTOUR will be described here. It takes as input a rectangular array F of real numbers that can be regarded as heights above a rectangular grid of nodes in the x-y plane. The second input, H, is one or several values for which one wants the level lines. The output, X, is a sequence of coordinates X,Y,X,Y,...,-1,-1,X,Y,... of points on the level lines. A pair of entries -1,-1 within X signals the start of a new contour line.

As always, there is the decision to be made whether the array F represents nodal values of a function F(X,Y) or values of H at positions in the array. CONTOUR does neither case: If F is nodal values of F(X,Y) one must first take the transpose of F, and if F is values of H above an array one must first reverse the order of the rows of F. Thus

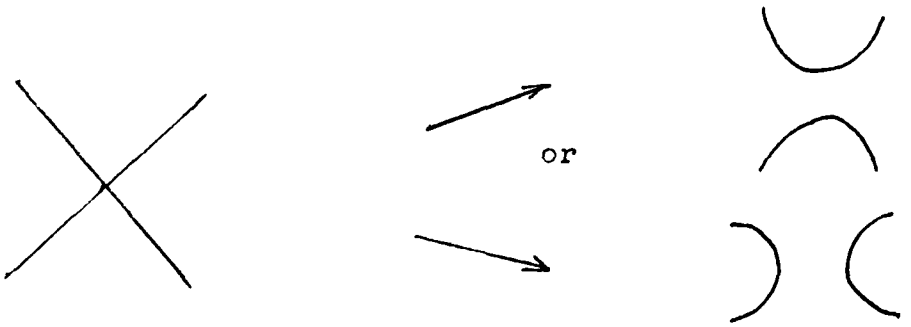
```
H CONTOUR ⍉ F          (function)
```

or

```
H CONTOUR ⊖ F          (array)
```

gives results in standard position with X increasing to the right, Y increasing upward.

Contour map algorithms require a check for F = H at a node of the array because the contour can have one or more branches there (e.g. at a saddle point). The interactive aspect of the function operates when such a point is found, and the user is required to enter a perturbed value of H. Then, at a saddle for example,



The function CHECK, called by CONTOUR, first checks F = H at corners (nodes of the array); then introduces centers of the rectangular patches where F is assigned the average of the values at corners; then checks F = H at centers. Given F ≠ H at any corner or center of the now triangulated array, no intersection or branching of contour lines is possible, and the only decision left is whether a contour is closed and lies within the array or open and terminates at an edge.

The function NODES finds nodes on the contour by linear interpolation of values of F at corners and centers. The output of NODES has four rows and as many columns as the number of nodes on the contour. The first two rows are encoded addresses of line segments that have a node, and the last two rows are coordinates of the nodes. Finally, the function SORT uses the encoded addresses to arrange the nodes in order, checks for open or closed contours, and returns the coordinates.

The logical structure of this algorithm is much simpler than others we have examined, and, even in APL, it is considerably faster than its FORTRAN competitors.

```
      ⎕CR'CONTOUR'
X←H CONTOUR F;N;S;E;W;⎕IO
⍝  X=X,Y,X... 1, 1... 1, 1...Y,X,Y
⍝  ON CONTOURS OF H=⍎F(X,Y) OR H=⊖F[I;J]
⍝  CALLS ∇X←CHECK H∇
⍝  INDICES - NORTH,SOUTH, EAST, WEST ON ⊖F
⎕IO←0
N←1+S←⍳(⍴F)[0]-1
E←1+W←⍳(⍴F)[1]-1
X←2↓CHECK H



      ⎕CR'CHECK'
X←CHECK H;Z;ZC
⍝  EXTERNAL VARS F,N,S,E,W,⎕IO=0
⍝  CALLS ∇X←SORT Y∇ AND ∇Y←NODES∇
⍝  CHECK AND GO - ANY NUMBER OF CONTOURS
→(1=⍴,H)/BGN
X←(CHECK H[0]),CHECK 1↓H
→0
BGN:Z←F-H
⍝  CONTOUR THROUGH A CORNER
→(∧/∧/⎕CT<|Z)/OK1
⍞←'F=',(⍕H),' AT A CORNER - TRY H='
H←⍎⍞
→BGN
⍝ CONTOUR THROUGH A CENTER
OK1:ZC←0.25×,+/Z[S;W],Z[S;E],Z[N;E],[1.5] Z[N;W]
→(∧/⎕CT<|ZC)/OK2
⍞←'F=',(⍕H),' AT A CENTER - TRY H='
H←⍎⍞
→BGN
⍝  NO CONTOUR
OK2:→(∨/∨/4≠|+/×Z[S;W],Z[S;E],Z[N;E],[1.5] Z[N;W])/OK3
⍞←'NO CONTOUR FOR H=',(⍕H),' - TRY H='
H←⍎⍞
→BGN
OK3:X←SORT⍎NODES
```

```
      ⎕CR'NODES'
Y←NODES;K;X;V
⍝  EXTERNAL VARS N,S,E,W,Z,ZC,⎕IO=0
⍝  ADDRESSES AND POSITIONS OF NODES
Y← 4 0 ⍴0
⍝  N-S NODES
NS:→(0=⍴K←(,(×Z[N;])≠×Z[S;])/⍳(1+⍴E)×⍴N)/EW
X←(6×X),X←⍉((⍴N),1+⍴E)⊤K
Y←Y,⍉X-¯3,0,(V↓(,Z[N;])[K]-V←(,Z[S;])[K]),[0.5] 0
⍝  E-W NODES
EW:→(0=⍴K←(,(×Z[;E])≠×Z[;W])/⍳(⍴E)×1+⍴N)/SW
X←(6×X),X←⍉((1+⍴N),⍴E)⊤K
Y←Y,⍉X-0,¯3,0,[0.5] V↓(,Z[;E])[K]-V←(,Z[;W])[K]
⍝  SW-C NODES
SW:→(0=⍴K←((×ZC)≠,×Z[S;W])/⍳(⍴E)×⍴N)/SE
X←(6×X),X←⍉((⍴N),⍴E)⊤K
Y←Y,⍉X-¯2,¯2,V,[0.5] V←V↓2×ZC[K]-V←(,Z[S;W])[K]
⍝  SE-C NODES
SE:→(0=⍴K←((×ZC)≠,×Z[S;E])/⍳(⍴E)×⍴N)/NE
X←(6×X),X←((⍴X)⍴ 0 1)+X←⍉((⍴N),⍴E)⊤K
Y←Y,⍉X-¯2,2,V,[0.5]-V←V↓2×ZC[K]-V←(,Z[S;E])[K]
⍝  NE-C NODES
NE:→(0=⍴K←((×ZC)≠,×Z[N;E])/⍳(⍴E)×⍴N)/NW
X←(6×X),X←1+⍉((⍴N),⍴E)⊤K
Y←Y,⍉X+¯2,¯2,V,[0.5] V←V↓2×ZC[K]-V←(,Z[N;E])[K]
⍝  NW-C NODES
NW:→(0=⍴K←((×ZC)≠,×Z[N;W])/⍳(⍴E)×⍴N)/0
X←(6×X),X←((⍴X)⍴ 1 0)+X←⍉((⍴N),⍴E)⊤K
Y←Y,⍉X+¯2,2,V,[0.5]-V←V↓2×ZC[K]-V←(,Z[N;W])[K]
```

```
      ⎕CR'SORT'
X←SORT Y;N;L;M;K;I
⍝  EXTERNAL VAR ⎕IO←0
X←⍳0
BGN:M←⍳L←(⍴Y)[0]
⍝  START WITH ZERO AND ONE OR TWO NEIGHBORS
N←¯1⍬0,(∨/(+/|I[L⍴0;]-I←Y[;0,1])∘.=2 3)/M
⍝  CHECK FOR ENDPOINT OR CLOSED CONTOUR
B:→(0=⍴K←(K≠N[1])/K←(∨/(+/|I[L⍴N[0];]-I)∘.=2 3)/M)/C
→(K≠¯1↑N←K,N)/B
→END
C:N←⌽N
⍝  FIND OTHER ENDPOINT
D:N←(K←(K≠N[1])/K←(∨/(+/|I[L⍴N[0];]-I)∘.=2 3)/M),N
→(0≠⍴K)/D
END:X←X,(,Y[N;3,2]),2⍴¯1
→(L≤⍴N)/0
Y←Y[(~M∈N)/M;]
→BGN
```


END
DATE
FILMED
6-80
DTIC

AD-A084 450 BROWN UNIV PROVIDENCE R I DIV OF APPLIED MATHEMATICS F/G 20/4
FINITE ELEMENT METHODS FOR HEAT TRANSFER PROBLEMS.(U)
APR 80 F BISSHOPP, R B CASWELL, M E MICHAUD DAAG29-76-G-0178
UNCLASSIFIED ARO -13548.2-MX NL

2 OF 2
AD A084450

END
DATE FILMED
10-80
DTIC

SUPPLEMENTARY

INFORMATION

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

ARO 13548.2-MX

AD-A084 450

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: addendum
2. GOVT ACCESSION NO.
3. RECIPIENT'S CATALOG NUMBER

4. TITLE (and Subtitle): Final report on manuscripts submitted or published under the subject Grant.

5. TYPE OF REPORT & PERIOD COVERED: Final Report 15 June 76 - 15 Sept. 79

6. PERFORMING ORG. REPORT NUMBER

7. AUTHOR(s): F. E. Bisshopp

(13)

8. CONTRACT OR GRANT NUMBER(s): DAAG29-76-G-0178

9. PERFORMING ORGANIZATION NAME AND ADDRESS:
Division of Applied Mathematics
Brown University
Providence, R.I. 02912

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: Project No. 1354M

11. CONTROLLING OFFICE NAME AND ADDRESS:
U. S. Army Research Office
P.O. Box 12211
Research Triangle Park, NC 27709

12. REPORT DATE: 12 July 1980

13. NUMBER OF PAGES: 4

14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)

15. SECURITY CLASS. (of this report): Unclassified

15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

DTIC ELECTE AUG 8 1980 S C

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

The view, pinions, and/or findings contained in this report are those of the author(s) and should not be construed as an official Department of the Army position, policy, or decision, unless so designated by other documentation.

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

80 8 1 030

FINAL REPORT

1. ARO PROPOSAL NUMBER: 13548M

2. PERIOD COVERED BY REPORT: 15 JUNE 1976 - 15 SEPT. 1979

3. TITLE OF PROPOSAL:

FINITE ELEMENT METHODS FOR HEAT TRANSFER PROBLEMS

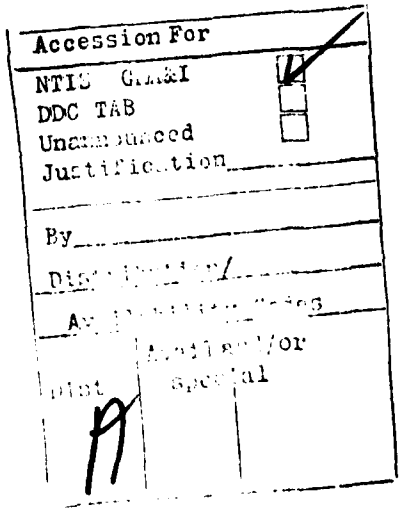
4. CONTRACT OR GRANT NUMBER: DAAG29-76-G-0178

5. NAME OF INSTITUTION: BROWN UNIVERSITY

6. AUTHOR OF REPORT: FREDERIC BISSHOPP

1354M

DR. FREDERIC BISSHOPP
DR. BRUCE CASWELL
BROWN UNIVERSITY
DIVISION OF APPLIED MATHEMATICS
PROVIDENCE, R. I. 02912



7. LIST OF MANUSCRIPTS SUBMITTED OR PUBLISHED UNDER ARO SPONSORSHIP DURING THIS PERIOD:

by FREDERIC BISSHOPP

1. A Lagrangian Finite Element Method for Barotropic Fluids. (long version) This will be rewritten and submitted for publication when the numerical work is completed.

2. A Lagrangian Finite Element Method for Barotropic Fluids. (short version) This was presented at the International Symposium on Innovative Numerical Analysis in Applied Engineering Science, Versailles (France), 23-27 May 1977.

3. Difference Analogs of Hamiltonian Systems. This will be submitted for publication upon completion of an application of the method that gives an efficient algorithm for construction of particle paths in two-dimensional flows.

4. Biased Differences for Advection/Diffusion Problems. (long version) Three working papers on this subject will be rewritten as one or two papers to be submitted for publication.

5. Biased Differences for Advection/Diffusion Problems. (short version) This was presented at the Gordon Conference on Fluids in Permeable Media, Plymouth, New Hampshire, June 1978.

6. Computer Graphics Software. This is now available in the APL software library at Brown University.

by MARION MICHAUD

1. Numerical Simulation of Reservoirs. Ph.D. dissertation, August 1979

2. Accurate Waterflood Simulation Using Biased Differencing and Selective Grid Refinement. Latest draft, March 1980. To be submitted for publication.

7. Continued from previous page.

by E. W. FLERI

1. A Lagrangian Finite Element Method. (probable title)
   Ph.D. dissertation in preparation

7a. MANUSCRIPTS SUBMITTED OR PUBLISHED DURING THIS PERIOD WITH PARTIAL SUPPORT UNDER ARO SPONSORSHIP:

by W. K. LYONS

1. The Single Conservation Law of Discontinuous Media.
   Ph.D dissertation, May 1980

by T. G. McKEE, Jr.

1. Instability and Bifurcation for Two-Dimensional Regions of Constant Vorticity. (possible title)
   Ph.D disseration in preparation.

8. SCIENTIFIC PERSONNEL SUPPORTED BY THIS PROJECT DURING THIS REPORTING PERIOD:

	Summer	Academic Year
6/76 to 5/77		
Bisshopp, F.	2 mo.	9 mo. (10%)
Fleri, E. W.	--	9 mo.
Michaud, M.	1 mo.	--
6/77 to 5/78		
Bisshopp, F.	2 mo.	9 mo. (10%)
Fleri, E. W.	2 mo.	9 mo.
Michaud, M.	1.5 mo.	--
6/78 to 8/79		
Bisshopp, F.	1 mo.	9 mo. (10%)
Fleri, E. W.	3 mo.	--
Michaud, M.	3 mo.	5 mo.
McKee, T. G	2 mo	9 mo.
Lyons, W. K.	--	5 mo.

8a. DEGREES GRANTED DURING THIS REPORTING PERIOD:

MARION CATHARINE MICHAUD, Ph.D., June 1980.

WILLIAM KIMBEL LYONS, Ph.D., June 1980

8b. DEGREES EXPECTED FOR WORK SUPPORTED BY ARO:

E. W. FLERI, Ph.D. (June 1981)

T. G. McKEE,Jr., Ph.D. (June 1981)

9. RESEARCH FINDINGS

A two-volume technical report, dated 2 April 1980 and submitted to the U. S. Army Research Office, contains brief descriptions of our research on numerical fluid mechanics in its first seven sections. More detailed descriptions of parts of that research are provided in six accompanying appendices.